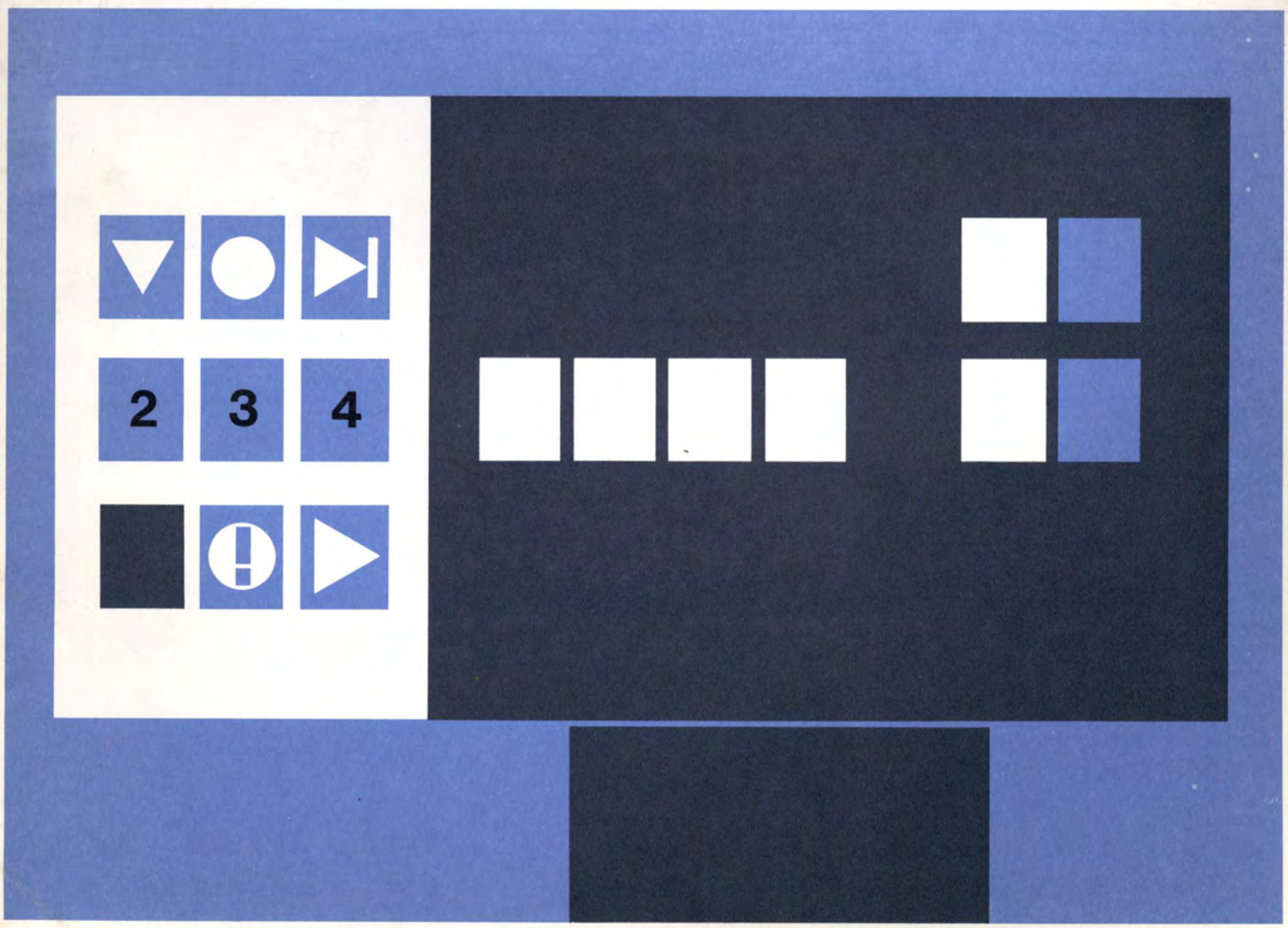
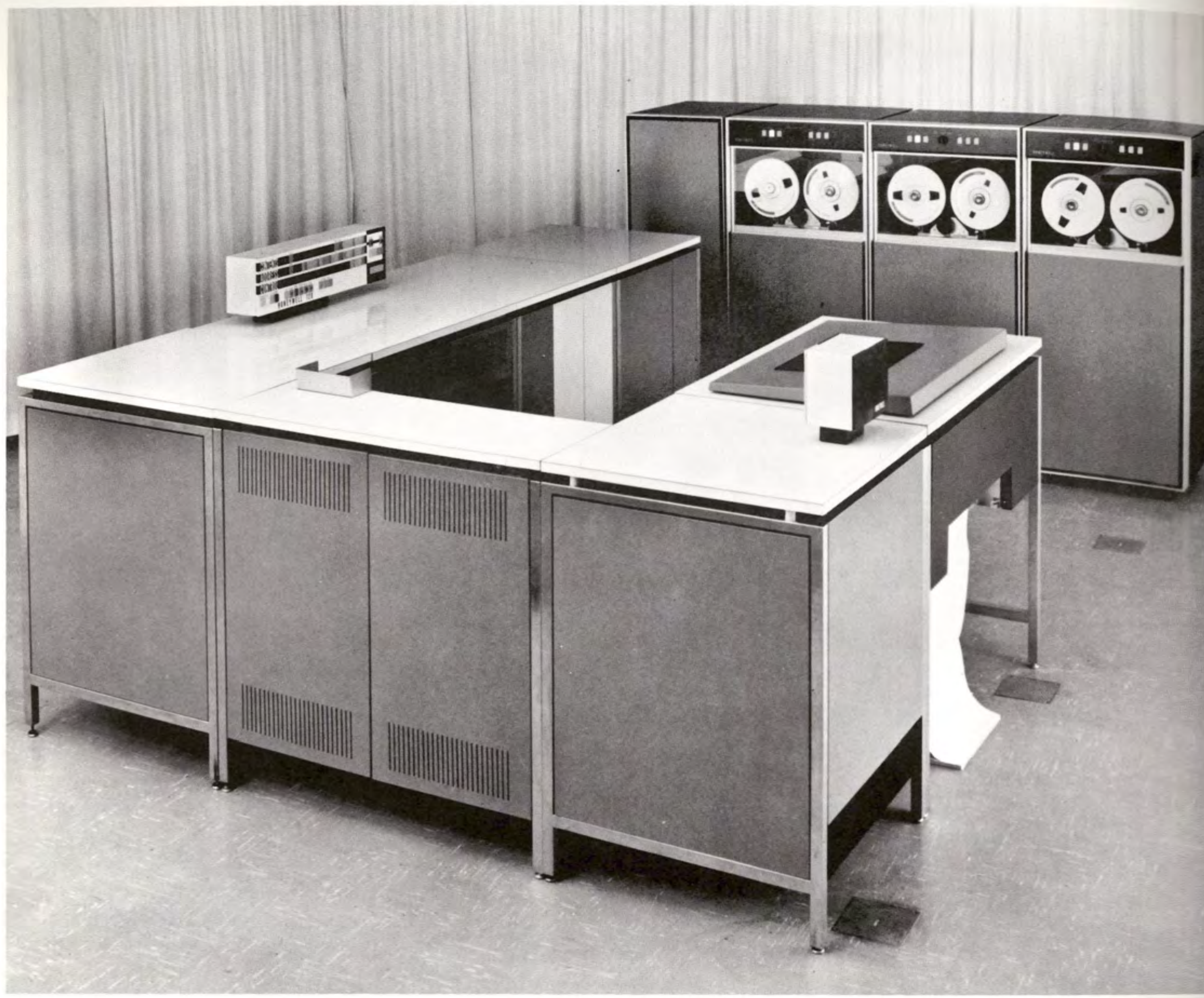
X6818.2013

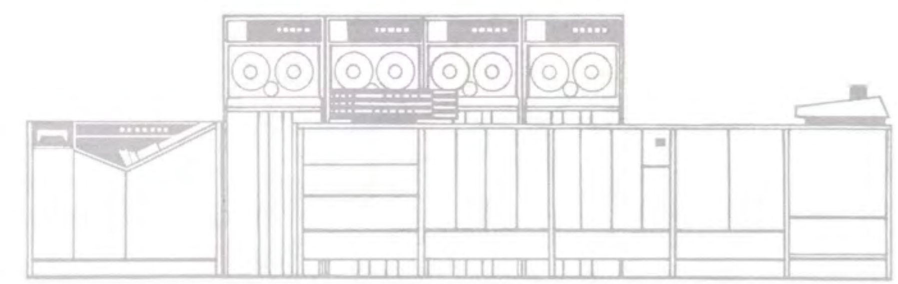
Honeywell

HONEYWELL SERIES 200
Summary Description





SERIES 200
SUMMARY DESCRIPTION



Honeywell
ELECTRONIC DATA PROCESSING

Printed in U.S.A.
NP—10734

* Underscoring denotes Order Number

143.0005.0000.3-012*
10M
5868

HONEYWELL

FIFTH EDITION

First Printing, December, 1967

Second Printing, August, 1968

Table of Contents

List of Illustrations

Foreword

This summary description of the Series 200 is intended for those having a general familiarity with data processing. Machine characteristics and programming aids are described in terms which should aid comparisons between the Series 200 systems and competitive equipment. The equipment characteristics reported herein remain subject to minor revision in order that design improvements may be incorporated.

1 Introduction

A computer, like any other tool used by man to tackle a task or problem, is limited in the extent to which it can be applied efficiently. A lightweight truck will probably break down if loaded considerably beyond its design limit. Likewise, a steam shovel is not the economic solution to digging postholes. A modular tool, however, can be applied to a wider range of jobs more efficiently. If the tool has several dimensions, each of which can in turn be modularized, the facility with which it can be tailored to handle specific jobs is enhanced even further.

This is how Honeywell has tackled the problem of matching computers to specific data processing requirements. By breaking computer capability into basic dimensions and providing a range of capability in each dimension, Honeywell is uniquely able to match a computer to a given job. Also, the computer can be expanded or modified very easily to match changes in system requirements. This approach to computer system design is the basis of Series 200.

Series 200 represents an "off the shelf" processing capability consisting of processing, input/output, and software modules that can be brought together in virtually any combination to form systems accurately tailored to solve any business or business-related data processing problem economically. Series 200 includes eight compatible processors which display outstanding cost/performance characteristics and offer the user great flexibility in his choice of speed, simultaneity, and memory capacity. A broad array of input/output devices, offered in several performance levels, provides many input/output media alternatives. Software, consisting of programming and application systems, is tailored to match the modularity of hardware.

The eight computers of Series 200 are the Models 120, 125, 200, 1200, 1250, 2200, 4200, and 8200. The 8200, the most powerful member of the series, is compatible not only with other Series 200 models but also with Honeywell's 800 and 1800 systems. Because of its dual processing ability, it is described in a separate publication entitled *Honeywell Series 200/ Model 8200 Summary Description,* Order No. 191.

The central processor of each computer model is referred to by a type number, e.g., the Type 1201 Central Processor of the Model 1200 computer. Note that any of three processors may exist in a Model 200 system: the Type 201, 201-1, or 201-2. All references to the Model 200 in this publication imply the Type 201-2.

Processing Dimension

- Memory speeds ranging from 3 microseconds per 6-bit character to 750 nanoseconds per four 6-bit characters
- Memory capacities ranging from 2,048 to 524,288 characters, in modular increments
- As many as 15 index registers for each resident program; flexible nanosecond control memories
- A universal set of powerful instructions affording program compatibility between processors
- Instruction and data compatibility with 1401, 1410, 1460, and 7010 systems
- Advanced programming and memory addressing methods, plus editing and multiply/ divide operations
- Powerful floating-point capability
- Extensive multiprogramming capabilities including memory address relocation

Input/Output Dimension

- Up to 16 peripheral operations performed simultaneously with computing
- Up to 48 peripheral control units connected to a processor; each accommodates one or several peripheral devices and is equipped with an automatic program interrupt facility
- A wide variety of peripheral equipment available in a range of performance capabilities, including communication devices, card equipment, magnetic tape and paper tape units, mass storage units, high-speed printers, banking equipment,

visual display devices, and memory-to-memory adapter units; also, controls for digital plotters, optical journal readers, a bill feed printer, and optical source-document readers

- Broad-scale real-time capability that includes an efficient interrupt facility, single- and multi-channel communication controls (the latter accepting data from up to 63 lines simultaneously), multi-level code handling, and a wide range of remote terminal facilities
- Either 6- or 8-bit information transfers between main memory and certain peripheral controls

Software Dimension

- Basic Programming System, which provides flexible software modules employing self-loading, unit-record techniques for use in small, card-oriented installations
- Operating System – Mod 1, which provides semi-centralized, automatic control for medium-scale tape or mass-storage installations
- Operating System – Mod 2, which provides completely centralized, highly automated computer management, with a minimum hardware overhead requirement, for medium- and large-scale installations of all types
- Operating System-Mod 2 (Extended), which incorporates all the facilities of Mod 2 with additional capabilities enabling multiprogramming and on-line, real-time communications, mass-storage support, and memory and file protection
- Application Systems, which assist directly in the performance of functions that are part of a user organization's operations

2 Processors

Series 200 processors are designed primarily for business applications and for jobs involving combined business processing, data communication, and scientific computing. In most data processing, the governing performance dimension is throughput – the quantity of data taken in, processed, and transferred to output media as computed results. High throughput requires not only an ability to transfer large quantities of data into and out of a processor; it also requires the capacity to process the data internally. This capacity includes the performance of all required computations and manipulations, while at the same time servicing demands from input/output devices quickly enough so that these devices can operate at their rated speeds. Therefore, the internal speeds of the processor must be high enough to allow the required combination of computing and input/output servicing. Clearly, then, high-throughput processors must possess a good *balance* of internal speed and potential input/output demand. As the following discussion will demonstrate, Series 200 processors incorporate an optimally balanced mixture of computing power and peripheral simultaneity at all levels of over-all throughput capability.

Computing Power

The ability of a processor to perform purely internal processing, involving only such operations as arithmetic, logical functions, data transfers, and editing, is largely a function of: (1) the amount of memory available for storing programs, as well as control and working data; (2) control and main memory speeds, which govern the time required to obtain and move instructions and data within the processor; (3) the selection of instructions in the processor's repertoire and the efficiency of the logic by which instructions are implemented; and (4) the memory addressing scheme used. Extended multiprogramming capabilities facilitate the maximum utilization of these factors. Series 200 processors provide computing power to meet the needs of any business jobs or applications involving business, scientific, and communication processing.

MEMORIES – SPEED AND CAPACITY: High internal speeds are assured by main memory cycle times ranging from 3 microseconds per 6-bit character to 750 nanoseconds per four 6-bit characters and control memory access times from 270 down to 125 nanoseconds. For example, consider the following statistics, which are based on typical situations:[1]

Operation	Execution Time, Microseconds Model 120	Model 1200	Model 4200
Decimal Add	69	35	13
Compare	57	29	10
Branch if Character Equal	36	18	7
Move Characters to Word Mark	54	27	9
Floating Multiply	n/a	50	20

[1] Three-character addresses are used to refer to 5-character operand fields. Instruction access times are included in the times shown. The times for floating multiply refer to operations using a 36-bit mantissa and a 12-bit exponent.

The speeds of Series 200 memories are complemented by the wide range of storage capacities available at each speed level. Memory size in the 121 processor ranges from 2,048 to 32,768 six-bit characters. At the other end of the scale, Type 4201 processors are available with 131,072- to 524,288-character memories. The modularity of Honeywell systems is exemplified by the relatively small increments in which main memory can be expanded, even at high capacity levels. High-speed control memories of from 13 to 57 control registers are used in all processors.

Information is stored in main memory locations either in pure binary form, as 6-bit alphanumeric characters, or as signed decimal quantities. Any number of consecutive locations can be grouped to form fields; groups of consecutive fields can be delineated as items.

MEMORY CAPACITIES AVAILABLE

PROCESSOR TYPE	MEMORY CYCLE TIME	2	4	8	12	16	20	24	28	32	40	49	57	65	81	98	114	131	163	196	229	262	327	393	458	524
121	3 microseconds/char.	✓	✓	✓	✓	✓	✓	✓	✓	✓																
126	2.5 microseconds/char.		✓	✓	✓	✓	✓	✓	✓	✓																
201-2	2 microseconds/char.		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓												
1201	1.5 microseconds/char.					✓				✓		✓		✓	✓	✓	✓	✓								
1251	1.5 microseconds/char.									✓		✓		✓	✓	✓	✓	✓	✓	✓	✓	✓				
2201	1 microsecond/char.					✓				✓		✓		✓	✓	✓	✓	✓	✓	✓	✓	✓				
4201	750 nanoseconds/4 chars.																	✓		✓		✓	✓	✓	✓	✓

Such groupings are defined by programmed or manual setting of punctuation bits associated with each memory location. (Fields and items are defined, respectively, by word marks and item marks.) Punctuation bits can also be set to form a record, which is defined as any unit of information that is to be transferred between main memory and a peripheral device as the result of a single peripheral data transfer instruction.

The Type 1201, 1251, 2201 and 4201 Central Processors possess an 8-bit transfer capability in addition to the standard Series 200 6-bit transfer. Certain specified peripheral control units can take advantage of this increased central processor flexibility. In the 6-bit mode, only the data portion of a character is transferred between main memory and the control unit. In the 8-bit mode, both data and punctuation are transferred between memory and the control unit. In this mode, record marks do not terminate the information transfer. Instead, transfers are terminated as specified by a count field in the information transfer instruction or by control characters associated with the control unit. This 8-bit transfer capability is available as Features 1120, 1121, and 1118 (Extended Multiprogramming and 8-Bit Transfer) for the Types 1201/1251, 2201, and 4201, respectively. This capability also requires that the Types 1201/1251 and 2201 be equipped with Storage Protection (Features 1114 and 1117, respectively).

There are no reserved input/output areas in main memory. The programmer has complete freedom in specifying the locations and sizes of such areas to meet the needs of any program. This allows both a high degree of programming flexibility and economical usage of memory.

A parity bit in each character position is used to maintain the accuracy of all data. Parity checking, performed automatically, is a method of checking a character each time it is moved in memory to insure that it retains its original value.

INSTRUCTIONS: Series 200 processors have available a repertoire of instructions which, with tremendous flexibility and power derived from the use of variant characters, can handle all arithmetic, logical, control, editing, and input/output functions necessary for business data processing. Also included in all processors are instructions for dealing with peripheral and communication interrupts and for manipulating data in codes of up to 12 levels. Hardware multiply and divide operations are standard in all processors except the 121 and 126. Types 1201, 1251, 2201, and 4201 can be equipped with a floating-point arithmetic facility for use in scientific applications.

The availability of Series 200 instructions in individual processors is shown in the accompanying table.

Instructions are variable in length. The basic instruction format consists of an operation code which specifies the type of operation to be performed, two operand fields which specify the binary addresses of fields to be used in the operation, and a variant character:

Operation Code	A Address	B Address	Variant Character

Series 200 Instruction Repertoire

Name of Operation	Processor Type[1] 121/126	200-2	1201/1251/ 2201/4201
Fixed-Point Arithmetic			
Decimal Add	S	S	S
Decimal Subtract	S	S	S
Decimal Multiply	–	S	S
Decimal Divide	–	S	S
Binary Add	S	S	S
Binary Subtract	S	S	S
Zero and Add	*	*	S
Zero and Subtract	*	*	S
Floating-Point Arithmetic			
Floating Add	–	–	*
Floating Subtract	–	–	*
Floating Multiply	–	–	*
Floating Divide	–	–	*
Store Floating Accumulator	–	–	*
Load Floating Accumulator	–	–	*
Floating Test and Branch on Accumulator Condition	–	–	*
Floating Test and Branch on Indicator	–	–	*
Decimal to Binary Conversion	–	–	*
Binary to Decimal Conversion	–	–	*
Store Low-Order Result	–	–	*
Load Low-Order Result	–	–	*
Binary Mantissa Shift	–	–	*
Binary Integer Multiply	–	–	*
Logical Functions			
Extract	S	S	S
Half Add	S	S	S
Substitute	S	S	S
Compare	S	S	S
Branch (Unconditional)	S	S	S
Branch on Condition Test	S	S	S
Branch on Character Condition, Basic	S	S	S
Branch on Character Condition, Extended	*	*	S
Branch if Character Equal	*	*	S
Branch on Bit Equal	*	*	S

Name of Operation	Processor Type[1] 121/126	200-2	1201/1251/ 2201/4201
General Control Functions			
Set Word Mark	S	S	S
Set Item Mark	S	S	S
Clear Word Mark	S	S	S
Clear Item Mark	S	S	S
Halt	S	S	S
No Operation	S	S	S
Change Addressing Mode	S	S	S
Change Sequencing Mode	*	*	S
Store Control Registers	S	S	S
Load Control Registers	S	S	S
Load Index/Barricade Register	–	–	*
Store Index/Barricade Register	–	–	*
Interrupt Control Instructions			
Store Variant and Indicators	S	S	S
Restore Variant and Indicators	S	S	S
Resume Normal Mode	S	S	S
Monitor Call	S	S	S
Data Move Instructions			
Move Characters to Word Mark	S	S	S
Load Characters to A-Field Word Mark	S	S	S
Extended Move	*	*	S
Move and Translate	*	*	S
Move Item and Translate	*	S	S
Move or Scan	–	–	*
Table Lookup	–	–	*
Editing			
Move Characters and Edit	*	*	S
Input/Output			
Peripheral Data Transfer	S	S	S
Peripheral Control and Branch	S	S	S

[1] S = standard, * = optional, – = not available.

The variants can be used to expand the meaning of the operation code or to specify literally a piece of data to be used in the operation. However, there are many times when not all of these instruction elements are needed, in which case they may be omitted to minimize both the amount of memory storage required and the time necessary to retrieve and execute an instruction. Peripheral control and input/output instructions have a slightly different basic format from the one described above. The most common Series 200 instruction formats are illustrated below.

Operation Code	A Address	B Address	Variant Character
Operation Code	A Address	B Address	
Operation Code	A Address	Variant Character	
Operation Code	A Address		
Operation Code	Variant Character		
Operation Code			

ADDRESSING: All Series 200 main memory locations are directly addressable. Three additional features facilitate advanced programming and addressing of large memories — indexed and indirect addressing and variable-length address interpretation.

A Series 200 processor may have a maximum of 15 main memory index registers available with each program resident in memory (see table below). These registers provide an automatic means for address modification without altering the instruction in which the address is modified. Indirect addressing enables the user to reference stored information via one or more intermediary addresses. Variable-length address interpretation refers to the ability of Series 200 processors to operate in three different address interpretation modes, allowing the programmer to code instructions using either 2-character, 3-character, or 4-character addresses. This facility provides the flexibility necessary to allow the direct addressing of large memories, while at the same time saving processing time and memory space when working in localized areas of memory.

SERIES 200 INDEX REGISTERS

PROCESSOR TYPE	MINIMUM	MAXIMUM
121	0	6
126	0	6
201-2	0	15
1201	15	15 available with each program resident in memory
1251	15	
2201	15	
4201	15	

MULTIPROGRAMMING: In a multiprogramming environment, it is necessary to have facilities to prevent active programs from altering information which is not relevant to their operation, to decide where new programs are to be loaded into main memory, and to allow certain high-priority programs to run without interruption. The Extended Multiprogramming and 8-Bit Transfer Feature provides the Type 1201, 1251, 2201, and 4201 processors with these facilities by including the following capabilities: storage protection with base relocation, interrupt masking, and instruction timeout.

The storage protection with base relocation capability consists of a relocatable, 2-fence barricade system. This system prevents the active program in main memory from writing into areas of memory which are not assigned to it. The lower fence is provided by the base relocation register and the upper fence by the index barricade register.

A Monitor program keeps track of the locations of the various programs stored in memory and can relocate the active area of memory to any number of 4096-character banks of memory. Interrupt masking gives the Monitor program the ability to mask out interrupts during the processing of high-priority programs (e.g., communications). The instruction timeout capability terminates a prolonged instruction extraction or execution which would prevent service to an interrupt within a required time interval.

Simultaneity

The speed of internal processing is one of the most important standards in evaluating the total throughput of a system; peripheral simultaneity is the other. Series 200 processors possess several features which enable them to provide powerful but easy-to-use simultaneity: program-assignable read/write channels, multiple device accommodations, and an interrupt processing facility.

READ/WRITE CHANNELS AND DEVICE ACCOMMODATIONS: The use of program-assignable read/write channels enables Series 200 processors to compute while concurrently servicing from 3 (Type 121 maximum) up to 16 (Type 4201 maximum) input/output operations. In addition, Series 200 processors provide interface facilities for a large number of peripheral controls, ranging from a possible 9, in the case of the 121, up to 48 in the 4201. The high internal speeds of these processors permit demands for service by peripheral devices to be satisfied even when the highest degree of possible simultaneity is implemented. Perhaps even more significant than the effect is the cause: this capacity is built into every Series 200 processor. It does not depend upon complex software or expanded system configurations.

The number of peripheral controls which a Series 200 processor can accommodate depends upon four factors: (1) the number of "unit loads" of power required by the controls to be connected; (2) the number of unit loads available from the processor; (3) the number of peripheral addresses required to connect the controls; and (4) the number of address assignments which the processor provides. Depending upon its identity, a peripheral control may require either one or two unit loads of power and from one to three addresses. The numbers of unit loads and address assignments available in each processor are shown in the accompanying table. Note that the number of available unit loads is expandable in all processors except the 126, 1201, and 1251. In the Type 121, 2201, and 4201 processors, the number of available address assignments can also be expanded.

INPUT/OUTPUT CHARACTERISTICS OF SERIES 200 PROCESSORS

PROCESSOR TYPE	ADDRESS ASSIGNMENTS Stand.	Max.	UNIT LOADS OF POWER AVAILABLE Stand.	Max.	NUMBER OF I/O OPERATIONS SIMULTANEOUS WITH COMPUTING Stand.	Max.
121	5	16	3	8	2*	3
126	16	16	8	8	2	4
201-2	16	16	8	16	3	4
1201	16	16	16	16	4	4
1251	32	32	32	32	6	6
2201	16	32	16	32	4	8
4201	32	48	32	48	8	16

* Card reading cannot be performed simultaneously with card punching in the basic 120.

The basic Model 120 is equipped with integrated peripheral controls for a 450-line-per-minute printer and for a 400-card-per-minute card reader and a punch that processes from 100 to 400 cards per minute. Also available is the Type 103 Magnetic Tape Control, one of which can be connected directly to the Model 120 peripheral interface; it accepts four 13,300-character-per-second magnetic tape units. Either of two optional features (Feature 1015 or 1016) allows the connection of up to five standard Series 200 peripheral controls in addition to those already mentioned (up to seven peripheral controls if the Type 103 control is not included in the system). The exact number of standard controls which can be added to the Type 121 processor varies according to the power and address requirements of the controls.

Data is transferred between main memory and a peripheral control via a read/write channel assigned by the instruction which initiates the transfer. Whenever an input/output operation is to be performed, a program-assigned read/write channel completes the path between the required peripheral device and the main memory.

In central processors other than the Type 4201, standard transfer rates of either 83,300 or 167,000 characters per second are associated with each read/write channel. Transfer rates higher than those attainable with a single read/write channel can be achieved by interlocking two or more read/write channels.

Rather than interlocking read/write channels, the Model 4200 traffic control offers a variable number of memory cycles per unit of time to each read/write channel, depending upon the read/write channel assignment code used in the instruction which initiates the operation. This technique provides read/write channel data transfer capacities ranging from 83,300 to 500,000 characters per second. Effectively, then, the Model 4200 incorporates variable-speed read/write channels. The most important advantage of this arrangement is that the read/write channels not made busy by a high-speed transfer are available for use in other operations.

The degree of peripheral simultaneity achieved by any Series 200 processor depends upon the number of read/write channels with which it is equipped. Standard equipment of a Model 120 processor includes three read/write channels, of which any two can be active at one time. The simultaneous use of the third channel is available with Feature 1016. Sixteen read/write channels are available in a Model 4200, allowing a like number of input and output operations, in any combination, to go on at the same time as internal processing. In order to appreciate the full power of the read/write channel concept, consider the following statistics. In one minute, a Model 200 system having four read/write channels can:

read 800 cards;

punch 10 columns of data into each of 400 cards;

print 1300 lines of 120 characters each;
read or write 4360 tape records of 500 characters;
(or perform any combination of four I/O operations) and in the same minute, execute 1.25 million instructions.

INTERRUPT PROCESSING FACILITY: The Series 200 automatic program interrupt facility provides simple but efficient supervision of processing involving combinations of input/output operations and computing. This facility allows automatic branching as necessary between a main program and servicing routines for all input/output devices. It obviates the need for programmed tests to detect the completion of input/output operations. The automatic hardware interrupt has important applications in the field of data communication and other real-time areas, but it is equally applicable to the supervision of operations as universal as reading and punching cards and paper tape, as well as reading and writing magnetic tape.

The Series 200 interrupt processing facility consists of a hardware program interrupt, which signals a particular condition in an input/output control unit, and a set of instructions used in processing interrupts. A program interrupt occurs whenever a peripheral device has completed an input/output operation. For example, an interrupt occurs at the end of data transfer in a tape read or write operation. Likewise, the receipt of a character from a remote station by a communication control may be signalled by a program interrupt. Interrupts from particular peripheral controls can be allowed or inhibited by a program as necessary.

A program interrupt is accompanied by: (1) automatic storage of main program indicator values and an indication of interrupt source; and (2) automatic branching to a routine whose address was previously loaded by program into a special control memory register. This routine can then proceed to determine the number and source of existing interrupts and to process the corresponding input/output demands according to whatever priority was specified by the programmer. The interrupt instruction subset is particularly helpful in this regard. After all demands have been processed, only a single instruction is necessary to resume the main program at its point of interruption and to restore all main program indicators to their previous values.

Conversion Compatibility

Series 200 processors are designed according to the Liberator concept, which allows the users of various competitive systems to take advantage of the superior performance of Honeywell systems without incurring the prohibitive costs of reprogramming. For example, the instruction repertoire of Series 200 processors is similar enough to those of several other processing systems, viz., the IBM 1400 series, to allow simple translation of programs written for these competitive systems to a form suitable for execution on higher-performance Series 200 systems. Compatibility of the larger processors — Types 1201, 1251, 2201, and 4201 — is further increased when operating in the "S" (Special) mode of processing.

Structural Modularity — Reliability

A major feature of the structural design of Series 200 processors is the use of integrated system modules. Each module contains all the circuitry required for a particular system function; for example, one module contains all the printer control circuitry, another contains the components of the arithmetic unit, etc.

This modularity greatly simplifies expansion of a system: in most cases, expansion involves little more than plugging in additional modules. The reliability of components within each module has been maximized through the use of silicon semiconductors. In addition, Series 200 takes advantage of the latest advances in the application of monolithic integrated circuits.

Exemplifying the advanced design of the Series 200 in verifying component reliability are the Peripheral Bus Switch and the Peripheral Control Tester on the Type 4201. These features permit any peripheral device and control unit to be selected and tested, for maintenance or troubleshooting purposes, without cycling down the central processor or interrupting the operation of other peripherals which are not under test. This capability is extremely important today in communication and real-time systems, where it is not possible to interfere with the operation of some components. The Peripheral Bus Switch is a manually-operated, integrated-circuit electronic switch which enables from one to four peripheral controls to be independently connected to or disconnected from a Model 4200 processor. The Peripheral Control Tester is a programmable device used to simulate a Series 200 central processor in testing the peripheral controls. Both the Peripheral Bus Switch and the Peripheral Control Tester are mounted in standard Series 200 logic cabinets.

Summary

Series 200 processors possess optimum combinations of high memory speeds, modular memory capacities, powerful instructions, efficient addressing methods, and flexible input/output traffic facilities which afford the computing power and simultaneity necessary for high throughput rates. The productivity of these processors is enhanced by their programming and operating simplicity. Basic hardware compatibility enables users of competitive systems to convert easily to take advantage of superior Series 200 performance. Sound hardware design, always a Honeywell plus, provides modularity and assures reliability. Processors are equipped with:

- Direct, indexed, and indirect addressing
- 2- to 4-character address interpretation
- Program-assignable read/write channels
- Automatic program interrupt
- 5-, 6-, 7- and 8-level code processing facility

Other processor facilities are tabulated below:

SUMMARY OF PROCESSOR CHARACTERISTICS

Processor Type	Main Memory Speed (cycle time)	Memory Capacity (thousands of characters)	Maximum Number of Peripheral Controls Accepted	Max. No. of I/O Operations Simultaneous with Computing	Advanced Programming Instructions	Financial Edit Instruction	Multiply and Divide Instructions	Scientific Processing Instructions	Memory Protect Facility	Extended Multi-programming Capability	8-Bit Transfer Capability
121	3 microseconds per character	2 - 32	9	3	*	*	—	—	—	—	—
126	2.5 microseconds per character	4 - 32	9	4	*	*	—	—	—	—	—
201-2	2 microseconds per character	4 - 65	16	4	*	*	Standard	—	—	—	—
1201	1.5 microseconds per character	16 - 131	16	4	Standard	Standard	Standard	*	*	*	*
1251	1.5 microseconds per character	32 - 262	32	6	Standard	Standard	Standard	*	*	*	*
2201	1 microsecond per character	16 - 262	32	8	Standard	Standard	Standard	*	*	*	*
4201	750 nanoseconds per four characters	131 - 524	48	16	Standard	Standard	Standard	*	*	*	*

* Feature optional.

— Feature not available on this processor.

3 Input/Output Devices

Series 200 includes a wide variety of input/output devices so as to enable the use of numerous input/output media. The following devices are offered: operator's consoles, magnetic tape units, card reading and punching equipment, random-access drum units, disk pack drives, disk files, printers, a paper tape reader, a paper tape punch, reader/sorters, multiple tape listers, teller terminals, Visual Information Projection units, and remote terminal devices (including the Data Station) and communication controls (discussed in Section 4). Also offered are control units for digital plotters, optical journal readers, a bill feed printer, and optical source-document readers. Most devices are offered in several performance levels, allowing the user to choose a desired input/output medium at an economical processing level. Particularly significant is the fact that all devices described here, except those for which limitations are specifically indicated, can be connected to any Series 200 processor, contingent only upon the availability of the requisite unit loads of power and address assignments (see page 11). The great flexibility thus provided allows accurate tailoring of system capabilities to satisfy user requirements.

Peripheral Controls

Peripheral controls are used to regulate the transfer of data between a processor and input/output devices. A significant feature is the fact that these controls operate independently of the central processor and require memory access only when information transfers are performed. In particular, all data validity checks, such as parity checks in magnetic tape transfers, are performed by the controls and do not involve the central processor in any way. Most peripheral controls can also generate an interrupt signal at the completion of data transfer. The initiation of the signal is allowed or inhibited by program control.

Controls for many of the Series 200 input/output devices can accommodate multiple devices. The other devices require their own individual controls.

The Type 121 and 126 processors include integrated controls for: (1) either a card reader/punch or, separately, a 400-card-per-minute or a 600-card-per-minute card reader and a card punch; and (2) a 450-line-per-minute printer. Also available for these processors are special controls for tape units which have nominal data transfer rates of either 13,300 or 26,700 characters per second. All other peripheral controls require a Series 200 Control Unit Adapter (Feature 1015 or 1016) in order to operate with the Type 121.

The Type 205 Magnetic Tape Switching Unit enables alternate connection of one or more tape units to different controls, except the Type 103 and 1038 Tape Controls used interchangeably with the Type 121 and 126 processors.

Type 220-1 Console

Control Panel and Consoles

A prospective Honeywell customer can choose one of three devices for over-all control of a Series 200 system: a control panel or one of two operator's console models. All devices provide a visual indication of system status and permit manual intervention into system operation.

The control panel, which is actually an integral part of the central processor, contains various control switches by which the operator can start and stop the machine and can load and interrogate both main and control memory locations. Sense switches may be used in conjunction with programmed instructions to stop processing or to select predetermined program paths, thereby increasing the flexibility of a program.

The Type 220-1 Console contains a console typewriter which may be used as a peripheral device, operating under program control, or as a logging typewriter. The central processor control panel is used in conjunction with the Type 220-1.

In the Type 220-3 Console, most of the control panel functions, including direct access to the central processor, are performed by means of the console typewriter. In addition, the typewriter can perform the peripheral and logging operations described above. The standard control panel is replaced in the Type 220-3 with a smaller version containing only the main power switches, sense switches, and certain check condition indicators. The control panel of the Type 220-3 Console may contain indicators used with storage protection and the additional sense switches used with the larger Series 200 processors.

Type 220-3 Console

Magnetic Tape Units

Three complete families of magnetic tape units are provided for use in Series 200 systems:

- Units which process ½-inch tape provide: (1) The standard means for storing 6-bit data; and (2) IBM compatibility, including end-of-file mark recognition and the ability to translate between IBM even-parity tape code and Series 200 processor code.
- Units which process ¾-inch tape provide data compatibility with Honeywell 400/1400/800/1800 systems.
- A unit which processes ½-inch, 9-channel tape provides data and reel compatibility with IBM ½-inch, 9-channel NRZ (nonreturn-to-zero) magnetic tape equipment.

Programmed tape operations include the following:

- ½-inch tape units – read forward, write forward, backspace one record, space forward one record, rewind, rewind and release (not available with 24-inch-per-second or 9-channel tape units), and erase; also available is read backward (except with the 9-channel tape unit).
- ¾-inch tape units – read forward, write forward, backspace one record, rewind, release, and regenerate tape channel.

As indicated in the accompanying table, data transfer speeds range from 4,800 to 144,000 characters per second for units processing ½-inch tape and from 32,000 to 88,000 characters per second for ¾-inch units. Also included in the table are "cross-gap" times, the presence of which points to a distinct advantage of Honeywell tape units. When a tape read or write operation is completed, the tape unit begins a deceleration interval which is coincident with the creation of part of the interrecord gap on tape. However, it is not necessary for the unit to stop before beginning to execute a new read or write operation. If such an operation is begun at any time during the deceleration interval, the unit merely accelerates, completes the interrecord gap, and begins the next operation.

The power of Series 200 peripheral simultaneity is evidenced by tape processing statistics: The proportion of available central processor time during a data transfer interval shared with a tape read or write operation ranges from 75% to more than 99%, depending upon the data transfer rate of the tape unit and the speed of the processor being used. Simultaneity is further increased in the case of ½-inch tape units: Reading and writing can proceed simultaneously under the direction of a single tape control at the same time that computing is in progress (the tape controls for 24-inch-per-second drives and the 9-channel tape drive do not have this facility).

The design of all Honeywell tape units incorporates the vacuum techniques which have earned an outstanding reputation for error-free operation. Vacuum control is used in mounting, driving, and stopping the tape so as to avoid any danger of damage; the reading surface of the tape has physical contact with the read/write head only. A write-enable ring and a manual tape unit switch guard information on tape from accidental destruction by an unintentional write operation.

All information written on ½-inch tape is immediately read and checked. During a write operation, a parity bit is generated for each frame and another is generated for each data channel. The parity bits accompany the data on tape. Frame and channel parity are checked while reading. Failure of any of these checks automatically causes an indicator to be set which can be tested by a programmed instruction.

Tapes written by IBM ½-inch, 9-channel (NRZ) tape equipment at 800-bpi density may be read on the Honeywell Type 204C-13 and 204C-14 tape units, and, conversely, tapes written by the 204C-13 and -14 may be read by the IBM units. This capability includes end-of-file mark recognition (tape-mark sensing).

An associated software package that complements the hardware provides data-interchange compatibility between the Honeywell and IBM 9-channel tape equipment. Control of input/output operations and other tape file-handling functions is provided by the software.

204
B-1, B-2, B-3, B-4, B-7, B-11, B-12, B-15, B-16, C-13, C-14

204
B-5, B-8, B-9

MAGNETIC TAPE UNIT SPECIFICATIONS

	Half-Inch Tape Units, Tape Transport Speed										¾-Inch Tape Units, Tape Transport Speed		
	24 in./sec.	36 in./sec.			48 in./sec.	80 in./sec.		120 in./sec.			60 in./sec.	120 in./sec.	
RECORDING DENSITY char./in.	200 556	200 556	200 556 or 800 800	800	200 556	200 556	200 556 or 800 800	200 556	200 556 566 800 or 800 800 1200 1200		533	533	740
TRANSFER RATE thousand char./sec.	4.8 13.3	7.2 20.0	7.2 20.0 or 28.8 28.8	28.8	9.6 26.7	16.0 44.5	16.0 44.5 or 64.0 64.0	24.0 66.7	24.0 66.7 66.7 96.0 or 96.0 96.0 144.0 144.0		32.0	64.0	88.8
REWIND SPEED inches/sec.	144	108	108	108	144	240	240	360	360		180	360	360
INTERRECORD GAP	.45″ .75″	.45″ .75″	.45″ .75″	.60″	.45″ .75″	.60″ .75″	.60″ .75″	.70″ .75″	.70″ .75″		.67″	.67″	.67″
CROSS-GAP TIME	18.7 ms 31.2 ms	12.5 ms 20.8 ms	12.5 ms 20.8 ms	16.7 ms	9.4 ms 15.6 ms	7.5 ms 9.4 ms	7.5 ms 9.4 ms	5.8 ms 6.3 ms	5.8 ms 6.3 ms		11.0 ms	5.5 ms	5.5 ms
TYPE NO.	204B-11, -12	204B-1, -2	204B-7	204C-13,-14	204B-15,-16	204B-3, -4	204B-8	204B-5	204B-9		204A-1	204A-2	204A-3

Card Equipment

In keeping with the concept of modular processing capability, Honeywell offers a flexible array of punched card equipment. The units described are recent Honeywell developments and include advanced card-handling techniques, such as column-by-column (end-feed) processing. End-feed card processing frees the central processor for other operations during a very high proportion of card equipment cycle time and enables the complete elimination of card cycle clutch points. In card devices employing clutch points, a card input/output instruction can only be acted upon at certain points in the cycle, a situation which limits the device's throughput. The use of end feeding in all Series 200 card equipment enabled Honeywell engineers to incorporate demand feeding, i.e., the execution of card input/output instructions *immediately* upon their receipt by the pertinent control unit. Demand feeding, in combination with the ability of Honeywell's card punch to accelerate over unused card fields, provides maximum rates for continuous card punching.

Automatic translation between standard 12-bit Hollerith card code and Honeywell central processor code is a standard facility on all Series 200 card devices. Transcription mode reading and punching are also available on all devices, except when connected to the integrated card control of a Model 120 or 125 computer. All card units have the ability to offset-stack selected cards under program control.

CARD READERS – 400, 600, 800, AND 1050 CPM

Four high-performance devices are offered for use in Series 200 systems to optically read 80- or 51-column punched cards: the Type 123, a 400-card-per-minute reader, the Type 123-2, a 600-card-per-minute reader (these two devices are available only with the Models 120 and 125 computers), the Type 223, an 800-card-per-minute reader, and the Type 223-2, a 1050-card-per-minute reader. The Type 223 (when the transcription mode capability is present) can also be equipped to read 90-column cards. Processed cards are sent to an output stacker, and those which fail data-protection checks can be offset-stacked under program control. End feeding substantially reduces the time normally required by edge-feed (row-by-row) readers for data transfer to and from main memory; therefore, other peripheral data transfers and computing can be performed during more than 99% of a card read cycle. Solid-state electronic components are incorporated in all card reader models to ensure optimum reliability. The speed, simplicity, and reliability of these devices combine to give them the best cost/performance ratios in the industry.

400-CPM CARD PUNCH

The Type 214-1 Card Punch operates at speeds of up to 400 cards per minute, depending upon which column is punched last. High-speed column skipping (Feature 066) is available as an option. This feature provides an automatic increase in card advance speed

SUMMARY OF CARD EQUIPMENT SPECIFICATIONS

	READERS				PUNCH	READER/PUNCH
SPEED cards/min	400 CPM	600 CPM	800 CPM	1050 CPM	100-400 CPM	100-400 CPM
SIMULTANEITY	The central processor, regardless of type, is free to perform other data transfers or computing during at least 99.4% of a card device's transfer interval.					
PROGRAMMED OPERATIONS	1. Read data and transfer to specified memory area. 2. Offset-stack card.				1. Punch data from specified area of memory. 2. Offset-stack card.	1. Punch data from specified area of memory. 2. Read data and transfer to specified memory area. 3. Read and punch same card. 4. Offset-stack card.
DATA TRANSFER [1] MODE	Automatic translation between Hollerith card code and 6-bit central processor code is standard. Additional transcription mode reading and punching capability also available.					
DATA PROTECTION	Illegal punch check Cycle check (including registration check)				Punch activation check	Illegal punch check and cycle check (including registration check) on reading; punch activation check on punching.
INPUT HOPPER/OUTPUT STACKER CAPACITY	3000/2500				1200/1300	1200/1300
TYPE	123	123-2	223	223-2	214-1	214-2

[1] Transcription mode reading and punching not available in a Model 120 or 125 computer's integrated card equipment control.

Type 223 Card Reader

when unused card columns are detected. This device also incorporates another new feature, dual-character punching, which employs a dual-die mechanism to punch two characters (columns) simultaneously and adds significantly to the high speeds and reliability of the unit. The Type 214-1 punch was designed for maximum reliability with minimum periodic maintenance. There are no cams, gears, or sliding parts in the feed mechanisms, making lubricating points completely unnecessary. The 214-1 enjoys the same simultaneity advantages afforded other Honeywell end-feed card devices: other processor and peripheral activities can occur during more than 99% of a card punch cycle. Punching errors are detected by a punch activation check; recognition of an error causes a program-accessible indicator to be set. The error card can be offset-stacked.

400-CPM CARD READER/PUNCH

This dual-purpose device (Type 214-2) actually has three operational modes; it reads, or punches, or reads a card and punches additional information into the card on the same pass. Punching speed ranges up to 400 cards per minute, depending upon which column is punched last. Operating speed is 400 cards per minute when reading only; if reading and punching during the same pass, the unit operates at its punching speed. This device combines all of the advanced features of the punch and readers described above. That is, the punch station employs dual-character punching, high-speed skipping (optional), as well as high reliability due to the absence of wear-producing cams, gears, and sliding parts. The reading station features optical techniques. Other peripheral data transfers and internal computation can be performed by the central processor during more than 99% of a card processing cycle. The reading and punching stations detect errors by means of illegal punch and cycle checks (including a registration check), and punch activation checks, respectively. When a discrepancy is sensed, a program-accessible indicator is set, and the card can be offset-stacked.

Type 214-2 Card Reader/Punch

Random Access Drum Storage and Control

The Honeywell Type 270A Random Access Drum Storage and Control provides a highly efficient, random access data storage medium for Honeywell computers. The drum subsystem achieves an optimum combination of high-speed access to large quantities of stored data and low storage cost per unit of information.

One to eight drum files can be connected to a control unit to operate on-line in a Series 200 system. The storage capacity of each drum is 20,480 records of 128 six-bit characters each, or 2,621,440 characters. Thus, a single control/drum subsystem can have a total capacity of over 20 million characters.

Program control of drum operations is maintained by use of two instructions: search-and-write and search-and-read. Both instructions can handle variable- as well as fixed-length records.

A drum rotation speed of 1140 rpm, coupled with the use of 512 read/write heads, provides access to a specified drum record in an average of only 27 milliseconds. Data transfer to and from the drum takes place at an average rate of 111,000 characters per second.

Only one memory cycle of central processor time is required for data transfer between the drum control and main memory. Therefore, the proportion of a data transfer interval available for other central processor operations varies from 67% for a Type 121 processor to 91% for a Type 4201.

Recorded data is protected by a 2-position PERMIT/PROTECT switch. When writing, the control unit generates cyclic coding and appends two cyclic check characters to each record. These characters are automatically checked while reading; any discrepancy results in the setting of a program-accessible indicator. A file protection feature (Feature 075) prevents the accidental programmed alteration of data encoded on any programmer-assigned group of 64 data tracks. As many as eight groups of tracks may be individually protected.

Type 270A Random Access Drum

Type 258 Disk Pack Drive

Disk Pack Drives

The Honeywell Disk Pack Drives are on-line storage devices which combine the features of magnetic tape (unlimited shelf storage) and magnetic disks (fast random access). Up to eight drives can be operated with a single control unit. Either the Type 257 Disk Pack Drive Control or the Type 260 Disk File Control can be used with the Type 258 (4.6 million characters) and the Type 259 (9.2 million characters) Disk Pack Drives. Data transfer rate for these devices is 208,333 characters per second. For systems not requiring the high data transfer rate of the Type 258 and 259, Honeywell offers the Type 259B Disk Pack Drive (requiring a 257B control), which has a data transfer rate of 147,500 characters per second and a capacity of 9.2 million characters. Available for use with the Type 201 and 201-1 Central Processors is the Type 259A Disk Pack Drive, which has the same capacity and data transfer rate as the Type 259B. The Type 257A Disk Pack Drive Control is required for the 259A drive. Disk pack drives are self-contained in a compact cabinet and consist of two main components: the disk pack and the access mechanism.

The control unit regulates the transfer of data between the central processor and the disk pack drive by providing temporary storage for each character transferred. The control also performs all checks on the data transfer operation and continually monitors the status of the disk pack drives. Up to seven seek operations can be performed simultaneously with a read or write operation by a single control unit. Data protection consists of a validity check for reading operations and verify reading and file protection for writing operations.

The Type 257-1 Disk Pack Drive Control and the Type 260 Disk File Control (with Feature 077) provide an 8-bit transfer capability. This capability allows data to be transferred between a Type 1201, 1251, 2201, or 4201 Central Processor and the disk control in the 8-bit transfer mode in addition to the standard Series 200 6-bit transfer mode. In the 6-bit mode, only data bits are transferred. In the 8-bit mode, both data and punctuation bits are transferred. The Type 257-1 control is identical in its operation to the Type 257 control with the exception of this 8-bit transfer capability.

Type 261 Disk File

Disk Files

The Honeywell Disk Files are fixed-disk storage devices which provide an extremely high on-line storage capacity for any Series 200 system. Up to eight Type 261 Disk Files, each with a capacity of over 150 million characters, can be operated with a single Type 260 Disk File Control. Up to four Type 262 Disk Files, each with a capacity of over 300 million characters, can be operated with a 260 control. Thus, the capacity of a single disk file subsystem may amount to over 1.2 billion characters. Any on-line data track is located in a maximum time of 120 milliseconds. Data is transferred at a rate of 190,000 characters per second.

The Type 261 consists of an access mechanism and 18 disks mounted on each of two side-by-side, vertical spindles. The Type 262 has two access mechanisms and 36 disks on each spindle.

The 260 control unit regulates and checks the data transfer as well as monitoring the status of the disk files. Up to seven seek operations can be performed simultaneously with a read or write operation by a single control unit. Data protection consists of a validity check for reading operations and verify reading and file protection for write operations.

The design of a disk file allows access to 1.2 million characters without read/write head movement. Another 1.2 million characters are available in 15 milliseconds. An 8-bit transfer capability, which is available when the 260 control with Feature 077 is attached to a Type 1201, 1251, 2201, or 4201 processor, allows data to be transferred in the 8-bit transfer mode in addition to the standard Series 200 6-bit transfer mode.

Printers

Honeywell offers printers to meet a wide variety of requirements. As indicated in the accompanying table, printing speeds offered range from 450 to 950 single-spaced lines per minute for alphanumeric characters and up to 1300 for lines containing a numeric character set; 96 to 132 print positions per line are available. Up to seven carbon copies can be provided.

Printing is performed in response to Peripheral Data Transfer instructions issued to the printer control from the central processor. The Peripheral Data Transfer instruction is also used to handle such functions as line and form spacing. A financial edit instruction allows the programmer to arrange output data into any desired format.

Type 222 Printer

SUMMARY OF PRINTER SPECIFICATIONS

	450-LPM PRINTER	650-LPM PRINTER	950-LPM PRINTER
SPEED ATTAINABLE WITH NUMERIC PRINT FEATURE	NA	1300 LPM	1266 LPM
% OF TIME AVAILABLE TO CENTRAL PROCESSOR FOR OTHER OPERATIONS DURING PRINT CYCLE	Type 121: 85% Type 4201: 96%	Type 121: 80-84% [2] Type 4201: 95%	Type 121: 73% Type 4201: 94%
PRINT POSITIONS PER LINE	120 or 132	96, 108, 120, or 132	120 or 132
CHARACTERS PER PRINT POSITION	63	63 standard; 49 with Numeric Print Feature (034, 035)	
SKIP SPEED inches per second	Up to 55 in/sec attainable as successive lines are skipped.		
VERTICAL SPACING	6 or 8 lines per inch.		
TYPE	222-5, 122[1]	222-1, -2, -3	222-4

[1] Type 122 Printer available only in Model 120 and 125 systems.
[2] Percentage is dependent upon the number of print positions per line.

During printing, an operator-changeable type roll on which characters are embossed moves past print hammers at each print position. Actuated as the proper character moves by, these hammers print the characters indicated by the print instruction. A cycle-check technique insures the accuracy of printed information. Standard drums for Honeywell printers have 63 alphanumeric characters available at each print position — 26 alphabetic, 10 numeric, and 27 special characters (e.g., credit symbol, asterisk, dollar sign, etc.). Each print position of the drum used for high-speed numeric printing has available a special 49-character set which is the same as the standard set except that it contains fewer special characters. Also available is a bar code drum that generates documents readable by the Type 289-8 Data Station Optical Bar Code Reader. Two special symbols of the standard print drums are replaced by the left and right ortho bars in the bar code drum.

The Type 229 Printer and Control is available to those customers owning or leasing Series 200 systems who qualify under "Educational Discount." The 229 printer can produce single-spaced copy at the rate of 400 lines per minute and vertically spaces lines at either 6 or 8 lines per inch. Any one of 56 characters can be printed in each of 120 print positions (132 print positions optionally available). Depending upon the paper stock being used, up to five carbon copies can be produced.

Paper Tape Equipment

The Honeywell paper tape reader (Type 209-2) processes 5- through 8-level tape at the rate of 600 frames per second; the punch (Type 210) operates at 120 frames per second. Reading and punching, as well as tape rewind and runout on the reader, are under program control. Tape stops within a frame's length at the end of a reading or punching operation, thus ensuring reliable reading of the first and last frames in a record.

Paper tape control units can be conditioned to process either codes of 5 and 6 levels or codes of 7 and 8 levels. This facility minimizes the amount of central processor time required for data transfer when processing 5- and 6-level tape.

Data transfer between the central processor and either the reader or the punch involves the central processor for only one memory cycle per 5- or 6-level

Type 209-2 Paper Tape Reader

frame and two memory cycles per 7- or 8-level frame. Thus, the central processor is free during more than 99.9% of a paper tape read or punch data transfer interval to perform computations and other input/output operations.

Frame parity can be generated by programmed instruction in preparation for punching. Likewise, frame parity can be checked by the program when reading tape. The 210 punch sets a program-accessible indicator when the end of tape is sensed.

Banking Equipment

Powerful computers and versatile input/output devices, plus a wealth of experience in the solution of many and varied banking problems, eminently qualify Honeywell to serve all types and sizes of banks.

READER/SORTER AND MULTIPLE TAPE LISTER

The availability of a magnetic ink character recognition (MICR) reader/sorter and a multiple tape lister and their controls ideally equips Honeywell systems to handle a full line of banking applications. The MICR reader/sorter operates at the speed of 1,560 documents per minute. The multiple tape lister, working in conjunction with the MICR reader/sorter, prints from 800 alphanumeric to 1600 numeric 22-column lines per minute.

TELLER TERMINAL EQUIPMENT

The Honeywell teller terminal equipment provides an on-the-counter, on-line banking system capable of processing all teller-assigned bank transactions. The equipment includes the Type 370 TELLER-REGISTER[1] Window Machine, which is used by the teller for all his bank transactions, and a remote transceiver that transmits transaction information between the Type 370 and the computer. Optional equipment for the system includes junction control units that provide for the connection of up to ten Type 370's to one transceiver, and off-line crossfooters and accumulators that permit operation of a Type 370 and a remote transceiver without the computer.

The Type 370 has four functional parts: the keyset, printer, document carrier, and supervisory control panel. The keyset includes data-entry keys (a nine-by-nine numeric keyboard), six operational status lights, 12 control-key/indicators, and 18 transaction keys. Within the Type 370 is a modular, removable printer which is designed to print on the journal tape and the customer's record simultaneously. A motorized feed/eject document carrier is provided for exact positioning of a passbook or form that is to be posted during a transaction. Located in a locked compartment just below the keyset is a supervisor's panel by which the teller supervisors can initialize the register.

The remote transceiver, operating as a link between the Type 370's and the computer, continuously scans the bid lines of all connected Type 370's. When a bid is detected, the scanning operation is halted and the transceiver becomes available to that particular Type 370 until a message is received, processed at the transceiver, sent to the computer, and returned. The message is read directly from the Type 370 to the computer by way of the transceiver; the transceiver does not store messages from the Type 370. Correct parity and control characters are added to the message at the transceiver before it is transmitted to the computer at line speeds. The reply message from the computer is stored in one of four transceiver buffers and is routed to the printer only after the complete message is received and has met parity requirements. As soon as the complete message is transmitted to the printer, resumption of transceiver scanning and the printing of the previously transmitted message occur simultaneously.

Type 370 TELLER-REGISTER Window Machine

Visual Information Projection Units

Cathode-ray tube (CRT) display units provide the facility for immediate access and display of stored data. They also introduce the capability to function

[1] Registered trademark of Bunker-Ramo Corporation.

DISPLAY STATION SPECIFICATIONS

TYPE	KEYBOARD	VIEWING AREA	DISPLAY CAPACITY
303	Separate from viewing unit—4-row, typewriter-like, alphanumeric—15 function keys—editing/special keys	7¾w x 5½h	Up to 768 characters
304	Same features as the 303, except that a Navcor electronic keyboard is used in place of the electromechanical keyboard of the 303	7¾w x 5½h	Up to 768 characters
311	15-key, block numeric—12 function/editing keys	4¾w x 3¾h	Up to 384 characters
312	43-key, numeric/block alpha—12 function/editing keys	4¾w x 3¾h	Up to 384 characters
317	Display device without keyboard	7¾w x 5½h	Up to 768 characters

off-line for message creation, verification, and correction by an operator, then on-line for message entry to and retrieval from the computer. The integration of computer and display provides a processing/communication system with the flexibility of localized query and update of data files, while maintaining centralized information control.

Type 303, 304, 311, 312, and 317 Display Stations are available, providing a wide range of display capabilities and keyboard arrangements for functional adaptability to any application. Depending upon the control unit storage available and the type of display used, input or output messages can consist of 32, 64, 128, 356, 384, or 768 characters. Typically, the number of display stations used may be from one to several dozen, although each station operates independently of the others. The information cycle from query to displayed response is completed in seconds whether the display-to-computer connection is a few hundred feet or, via communication lines, several thousand miles.

Display stations connected in any one area utilize a Type 323 Universal Control as a local storage medium, control, and communication link. For more diversified applications, terminal devices such as page printers and paper tape readers and punches can also be attached to a Type 323 control through an appropriate control module. Additional expansion, control, and communication modules can be connected to the local control to provide more buffer storage, editing functions, and communication capabilities.

Type 303 Display Station

The Type 323 Universal Control can be remotely or directly connected to a Type 281 or 286 Communication Control. With either remote or direct connection, communication modules are available to provide asynchronous or synchronous transmission of USASCII-coded data at rates of 1200, 2000, or 2400 baud over standard half-duplex communication lines.

Digital Plotters

The Type 234 Plotter Control enables a Calcomp[1] Series 500 Plotter (Model 563, 564, 565, or 566) to operate on-line with any Honeywell Series 200 central processor. The Calcomp Series 500 Plotters create very accurate ink-on-paper plots, producing curves and symbols of any desired shape. They operate digitally, moving the pen in discrete increments in any of eight directions with the pen either raised or lowered. They are used to advantage in situations where graphical or pictorial representation of computer data is easier to interpret and use than extensive numeric or alphabetic listings.

A complete plot may be executed by a single instruction. The maximum plotting speed is 300 increments per second, in any of eight directions. The plotter control employs the standard interrupt capability of the Series 200 processors.

Optical Journal Readers

The Type 235 Optical Journal Reader Control allows a National Cash Register (NCR) Model 420 Optical Journal Reader to operate on-line with any Series

[1] Registered trademark of California Computer Products, Inc.

200 central processor. The Model 420 optically reads printed journal tapes from a variety of input machines (e.g., cash registers, accounting machines, adding machines). The NCR font character set, consisting of 16 stylized numbers and symbols, is used.

The control performs no translation, but, instead, character equivalents have been established between the NCR font character set and the Honeywell code configuration. Under control of the central processor program, the 235 control transfers data from the reader memory to main memory at a rate of 83,333 characters per second. Standard Series 200 interrupt capabilities are included in the control unit.

The Model 420-1 Optical Journal Reader has a maximum speed of 26 lines per second, and the Model 420-2 has a maximum speed of 52 lines per second. Depending on the operating mode selected, unreadable lines will cause the reader either to halt after a predetermined number of backups and rereads or to mark the error line on the reverse side of the journal tape and continue the reading sequence.

Bill Feed Printer

The Type 237 Bill Feed Printer Control is a device which controls a Model 1404 Alphanumeric Printer and enables the printer to operate with any Honeywell Series 200 central processor. Printing, under control of the Type 237, can be performed either on continuous forms that are from 3½ to 18¾ inches wide or on cut-card forms of any size in the range between the standard 51-column card and the standard 80-column card. An 80-column card with an 80-column stub can also be processed.

When printing is performed on continuous forms, a printing speed of 600 lines per minute (single spaced) of alphanumeric information can be realized. Information is printed ten characters per inch over a 13.2-inch print line, a total of 132 character positions per line. Characters printed are .093 inches high; vertical spacing can be set at either six or eight lines per inch.

An outstanding capability of the Type 237 control and 1404 printer combination is the ability to print at a rate of 400 cards per minute per card feed when only the top line of the card is printed. Since there are two card feeds available, an effective printing rate of 800 cards per minute is realized.

An integral feature of the Type 237 control is its extensive error-checking ability. In addition to the several electromechanical checks performed by the printer unit, the 237 control performs four printing and decoding (cycle check) error checking operations, a validity check, and a comparison check. These checks ensure that printing takes place in the proper print position at the right time and that synchronization is maintained between the several operations required to produce printing.

Optical Readers

The Type 238 Optical Reader Control enables a 1287 Optical Reader (Model 1 or 2) to be connected to any Honeywell Series 200 central processor.

The 1287 Optical Reader is an on-line input device that uses optical scanning and electronic character recognition circuitry to read data from source documents into a Honeywell Series 200 computer equipped with the Type 238 control. The optical reader can read numeric handwriting, machine-printed numeric characters, optical mark-read numeric characters on paper or cut-card form documents, and cash register and adding machine tapes.

The Type 238 Optical Reader Control has its greatest application in the retailing industry for the processing of sales checks, merchandise orders, bills, and other original business documents. A Honeywell Series 200 system equipped with the Type 238 Control and the 1287 Optical Reader is also suited for use in many other industries that have previously realized only a portion of their data processing potential.

4 Data Communication Facilities

Honeywell provides a broad-scale data communication capability, the highlights of which are:

- Single-channel and multi-channel control units to handle an exceptionally wide array of communication lines, speeds, and terminal devices.
- Fast-access mass storage devices.
- Powerful processor communication features, including an automatic interrupt system, 5-through-8-level code handling capability, a programmable real-time clock, a standard interval timer, and a low-cost, programmable interval timer.
- An advanced, multi-purpose remote terminal device, the Data Station.
- A full line of software for interrupt processing and message handling.

This entire communication capability is available for use in any Series 200 system, regardless of the processor model chosen by the user.

Processor Communication Features

Several features available in Series 200 processors make them especially well suited to handle communication applications. The simultaneity, high internal speeds, and automatic hardware interrupt facility of these central units provide a very significant capability – *effective processing of communications and conventional jobs at the same time.* Flexibility in application design is provided by the ability to process USASCII as well as other 5-, 6-, 7-, and 8-level codes.

SIMULTANEITY

The use of program-assignable read-write channels in Series 200 processors enables them to direct the data flow to and from several peripheral devices and, concurrently, to perform computing operations. For example, a Model 200 processor can perform up to four input/output operations at the same time that internal processing is going on. Projected to a Model 4200 processor, this facility allows 16 peripheral data transfers to proceed simultaneously with computing. This greater throughput is simultaneity's chief contribution to integrated communication/business data processing systems.

INTERNAL SPEED

Concurrent I/O and internal processing must be coupled with internal speeds high enough to allow efficient handling of data received or transmitted to provide an effective computing system. Honeywell memory cycle times ranging from 3 microseconds per 6-bit character down to only 750 nanoseconds per four 6-bit characters provide internal processing speeds which are suitable complements to the simultaneity of Series 200. These speeds enable complete processing of communication data even when transmission is at high-volume rates.

AUTOMATIC INTERRUPT FACILITY

Standard equipment on all Series 200 processors is a completely automatic program interrupt facility. The advantage of this interrupt is that it enables simple but efficient direction of processing involving concurrent real-time and business or scientific applications. The interrupt facility allows automatic branching, as necessary, between a main program and real-time service routines. In particular, the readiness of a communication control to receive data for transmission or to relay data coming in from a line can automatically trigger entrance to a stored routine to service the external demand immediately. (Interrupt routines, applicable to most communication environments, are provided by Honeywell as part of the standard software.) Automatic signalling of control status obviates the necessity for programmed tests of these units to detect the arrival of data or the readiness to transmit. The interrupt facility also includes automatic storage of main program indicator values, as well as an interrupt source indication.

MULTI-LEVEL CODE-HANDLING FACILITY

All Series 200 processors are capable of bringing into memory and manipulating data in many different codes. This feature includes the ability to translate automatically between character codes of up to 12 levels.

TIMING DEVICES

Three types of devices are available for use in Series 200 processors to give programs access to real-time information. The Type 213-3 Interval Timer provides automatic program interrupts at program-specified intervals. The Type 213-4 Time of Day Clock permits a program to determine the current clock time in hours, minutes, seconds, and tenths of seconds. The Type 285-T Communication Interval Timer generates program-controlled interrupts in a Series 200 central processor via the Type 286 Multi-Channel Communication Control. Multiple 285-T timers can be connected to a 286 control. These devices may be used in such applications as: (1) timing of program runs; (2) logging times of remote inquiries and information input; and (3) starting programs at specified intervals or clock times, as in polling a communication network.

Data Communication Controls

Single-channel and multi-channel controls are available to enable Series 200 systems to receive and transmit data over toll and leased lines. One of the most outstanding features of these devices is the broad selection of lines, speeds, and terminal devices to which they can be connected – this selection is one of the largest offered by any manufacturer. The compatible services and equipment are indicated in the accompanying table. Combinations of some Type 285 adapters are available as Multi-Adapter Units.

SINGLE-CHANNEL CONTROLS

The Type 281 Single-Channel Communication Controls direct the transmission and reception of messages in 5- to 8-level codes at rates of up to 5,100 characters per second, depending upon what common carrier line is selected. The controls are distinguished from one another only by the lines and services to which they are connected; functionally, they are similar. These controls are half-duplex devices; i.e., messages are both transmitted and received, but not simultaneously. Additional controls can be added to a system in order to provide full-duplex or multi-channel operation.

MULTI-CHANNEL CONTROLS

The Type 286-1, -2, and -3 Multi-Channel Communication Controls direct the transmission and reception of message characters over as many as 63 half-duplex communication lines. The 286-1 services up to 3 lines, the 286-2 services from 4 to 15 lines, and the 286-3 can handle from 16 to 63 lines. A communication adapter is required to interface between the control and each line being used. The processor is interrupted as each character of a message is transferred. Data can be transferred at rates up to 300 characters per second in a single line.

MESSAGE-MODE CONTROLS

The Type 286-4 and 286-5 Message-Mode, Multi-Channel Communication Controls direct the transmission and reception of whole messages over as many as 63 half-duplex lines. The 286-4 services from 2 to 32 lines, and the 286-5 services up to 63 lines. A communication adapter is required as the interface between the control and each line being used. The total throughput capacity (all lines) of either control is 7,000 characters per second. Line-scanning priorities are established under program control.

Initially, a main memory input/output area address is stored in the message-mode control for each line connected to the control. When a character is received (or is to be transmitted) over a line, the address corresponding to that line is transferred directly and automatically to the central processor. The message character is then transferred to/from the main memory address indicated. The address of the next character position to be retrieved from main memory is always stored in the message-mode control, so that the next message character is once again transferred automatically. The processor is interrupted only when an entire message (or a designated portion thereof) has been transferred over any one line.

DATA PROTECTION

The validity of data being communicated is protected by three different means:

- A transmission lapse results in the automatic setting of a program-accessible indicator in the receiving processor.
- Where applicable, the controlling program can initiate an automatic message-receipting system.
- For codes of more than six bits, a frame parity check is available. A long-check feature is also available.

APPLICABILITY OF HONEYWELL COMMUNICATION EQUIPMENT

Terminal	Service & Line	Dataset[1]	Transmission Speed	Single Channel Control Type	Adapter Unit Type[2 3]
DATASPEED[4] 1 and 2	Voice-grade Line	202C-D	105 cps	281-1H	285-1H
DATASPEED[4] 5	Voice-grade Line	402C	75 cps Parallel Send	——	285-3A
DATASPEED[4] 5	Voice-grade Line	402D	75 cps Parallel Receive	——	285-4A
Honeywell Series 200 Computer. Other selected computers;	Voice-grade Line	201A-B	250/300 cps	281-2B	285-2B
other selected high-speed devices.[5]	Telpak A	301B	5100 cps	281-2F[6]	——
Honeywell Data Station	Voice-grade Line	202C-D (201A-B)[9]	120/300 cps	281-1M	285-1M
	Direct Connection (50 ft. maximum)	——	120/300 cps	281-1M	285-1M
Honeywell Type 370	Direct Connection (200 ft. maximum)	——	120 cps	——	285-1PD
	Voice-grade Line	202C-D	120 cps	——	285-1PM
Honeywell Visual Information Projection Equipment	Voice-grade Line	201A-B	250/300 cps	281-2R	285-2R
	Direct Connection (50 ft. maximum)	——	250/300 cps	281-2R	285-2R
	Direct Connection (200 ft. maximum)	——	250/300 cps	281-2S	285-2S
	Voice-grade Line	202C-D	120 cps	281-1R	285-1R
	Direct Connection (50 ft. maximum)	——	120 cps	281-1R	285-1R
	Direct Connection (200 ft. maximum)	——	120 cps	281-1S	285-1S
Honeywell Audio Response System (Tel. Co. Telephone Handset with TOUCH-TONE Key Pad/Card Dialer)[11]	Voice-grade Line	403A	Up to 10 cps in; speech out	——	285-8
IBM 1050	Voice-grade Line TWX C.E. (Prime)	103A	14.8 cps	281-1E	285-1E
	Voice-grade Line (AT&T) AT&T 150-Baud Line W.U. 180-Baud Line	103F 816 W. U. Type 1601A or Type 1181.1A	14.8 cps	281-1K[7 10]	285-1K[7 10]
IBM 7701, 7702, 1013	Voice-grade Line	201A-B	250/300 cps	281-2D	285-2D
	Voice-grade Line	202C-D	120 cps	281-2A	285-2A
TTY 19, 28	5-level Telegraph	TTY Circuit	60-100 wpm	281-1C[7]	285-1C[7]
TTY 32	W. U. Telex	W. U. 11671 or W. U. 11903A	66 wpm	281-1A[8]	285-1A[8]
TTY 33, 35	Voice-grade Line TWX C.E. (Prime)	103A	100 wpm	281-1B	285-1B
TTY 35	8-level Telegraph	103F	100 wpm	281-1D[7]	285-1D[7]
TWX Subscriber Stations	TWX	811B	60-100 wpm	——	285-1N

[1] Unless otherwise specified, data set designations refer to Bell System data sets or their equivalent.
[2] References to adapters imply Type 286 communication controls, since a 285 adapter interfaces each line connected to a multi-channel control.
[3] When a Type 285 is connected to the remote terminal via DDD, TWX, or TWX C.E., calls may be originated by either (1) manual dialing or (2) automatic computer control and operation via an associated Type 285-5A adapter and a Tel. Co. Automatic Calling Unit 801A or 801C.
[4] Trademark of American Telephone & Telegraph Co.
[5] Contact Honeywell Sales Representative for compatibility with specific computer and terminal arrangements.
[6] Type 281-2F single-channel control in Honeywell-to-Honeywell service is available in half-duplex and full-duplex forms.
[7] Private line use only.
[8] Both manual and automatic dialing capabilities are available.
[9] Data station with a Type 288-3 control and a Type 289-9 printer requires a 201A or 201B dataset and provides a transmission speed of 250/300 cps.
[10] Contact Honeywell Sales Representative for approval of this unit.
[11] Input via Tel. Co. TOUCH-TONE dial, manual key pad, or card dialer. Output–Voice (audio from computer).

Note: Parity Check is available at no extra cost on all single-channel controls, is available as Feature 086 on the Type 286-1, -2, and -3 Multi-Channel Communication Controls, and is standard on the Type 286-4 and -5 Message-Mode, Multi-Channel Communication Controls. (Controls and adapters carrying the suffix 1A, 1B, 1C, 1D, or 1H do not use this check.)

Long Check is available as Feature 087 on all 281-2 single-channel controls and the Type 286-1, -2, and -3 Multi-Channel Communication Controls. It is standard on the Type 286-4 and -5 Message-Mode, Multi-Channel Communication Controls. (Type 285-1 adapters do not use this check.)

COMMUNICATION SWITCHING UNITS

Communication switching units of the 215 series enable any two Honeywell systems to use alternately a group of communication lines or to switch between different groups of lines. This is accomplished by the switching of a group of 285 adapters between 286 controls. Up to 63 lines may be switched simultaneously.

Fast-Access Storage

A major requirement of many communication applications, such as those involving inquiry and message switching, is fast access to information which has been placed in storage. Of course, main memory provides the fastest access possible. But when dealing with large files, core memory becomes too expensive. To fill this need for economical storage, Honeywell offers the complete line of magnetic tape units and mass storage devices described in the preceding section. Magnetic tape units, available in a broad range of speeds and recording densities, offer a sufficiently fast medium for most applications at a very low cost in terms of storage capacity. Larger capacities and higher transfer rates are provided by the disk devices. They range in capacity from the 4.6 million characters of the Type 258 Disk Pack Drive to over 300 million characters with the Type 262 Disk File. Seek times as low as 15 milliseconds meet the requirement of fast access to the data.

Visual Information Projection Units

The many unique features of Visual Information Projection units provide extended flexibility to Honeywell's communication facilities. The units are ideally suited for businesses requiring instantaneous visual access to data stored in computer files. Used as the primary terminal device, display stations can either replace or supplement other types of input/output terminals such as page printers and paper tape devices. Designed to operate with any Series 200 processor, practically any number of display stations may be used to form an integrated processing/communication network. Regardless of the number of display stations in the network, however, the entire cycle from inquiry to appearance of the reply on the screen is virtually instantaneous.

FLEXIBILITY

All models of the display stations incorporate a wide range of display capabilities and keyboard arrangements in order to provide functional adaptability to any application. Options such as designation and usage of function keys and editing capabilities are also available with all models.

TRANSMISSION CHARACTERISTICS

All models of the display stations can operate at line speeds of 1200, 2000, or 2400 baud. An 8-bit code (7-bit USASCII plus parity) is used for synchronous transmission and a 10-bit code (7-bit USASCII plus parity and start and stop bits) for asynchronous transmission.

Data Station

The Honeywell Data Station is a multi-purpose remote terminal device which can be used for a broad range of communication applications, as well as for off-line jobs. This device gives branch offices, warehouses, remote reporting locations throughout a plant, or any other company outposts, the power to prepare source data locally and to communicate directly with a centrally located computer.

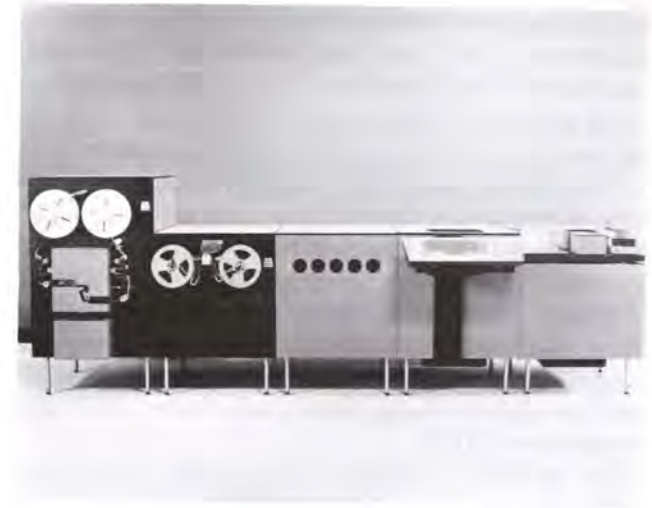
The Data Station features transmission speeds of 120 characters per second with the Type 288-1 Control Unit and 300 characters per second with the Type 288-3 Control Unit. A wide choice of input/output devices includes paper tape and punched card equipment, a keyboard, page printers, and an Optical Bar Code Reader that introduces new applications possibilities. A control unit can handle a keyboard plus various combinations of the other peripheral devices included in the accompanying table. The only restriction is that if the 289-9 printer is part of the system, the 288-3 control must be used and, vice versa, if the 288-3 control is used, the 289-9 printer must be included in the system.

DATA STATION PERIPHERAL DEVICES

Device	*Speed*	*Type*
Remote Line Printer	400 lines/min	289-9
Card Reader	120 char/sec	289-7
Paper Tape Reader	120 char/sec	289-4
Paper Tape Punch	120 char/sec	289-5
Optical Bar Code Reader	50 char/sec	289-8
Page Printer and Keyboard	10 char/sec	289-2
	40 char/sec	289-3
Keyboard	10 char/sec	289-2A

ON-LINE OPERATION

The Data Station transmits over a telephone-grade line using a DATA-PHONE[1] 202C or 202D data set with the 288-1 control or a 201A or 201B data set with the 288-3 control; party line operation is available. It uses the 7-bit USASCII code with parity included in an eighth bit. Communication can be directed locally by the operator or remotely by the central computer. Parity and long checking are used to protect against transmission errors. An automatic facility is available for initiating the retransmission of data in which errors have been detected.



Data Station

OFF-LINE OPERATION

When not being used for on-line operations, the Data Station can be used for local activities such as data preparation and editing. By way of illustration, possible off-line operations include keyboard-to-paper-tape and card-to-paper-tape with simultaneous printing.

Audio Response System

The Honeywell Audio Response System combines the high speeds of Series 200 computers with the control of prerecorded voice messages to provide a versatile method of placing the human user "on-line" with a data file. Applications of the system range from the simplest – data collection with no more than a "message received" tone playback – to complex audio message compositions when a hard-copy record is not required at the remote terminal. The system accepts inquiries from multiple remote Touch-Tone[1] telephones, composes combinations of words or phrases under central processor control, and transmits replies in the form of audio tones, words, phrases, or combinations of these. Replies which require more than one word or phrase have slight pauses between words, which are barely discernible to the listener. These pauses allow wide latitude in the choice of vocabulary for the recordings and the assembly of messages using this vocabulary.

[1] Registered trademark of the Bell System.

SYSTEM CONFIGURATION

The most basic system consists of a Type 286 Multi-Channel Communication Control and the Type 285-8 Touch-Tone Adapter. The 285-8 adapts up to six lines to any Series 200 central processor. Each line must be terminated with a 403A data set supplied by the Bell System. The electrical interface is between the 285-8 and the 403A, the data set which is compatible with remote Touch-Tone sets.

Expansion of the basic system is provided in two forms. The number of lines operating independently and simultaneously can be expanded to a maximum of 60 by adding one Type 285-8A option for every two additional lines. The other expandability dimension is the type of audio responses that are available with the addition of the following options:

1. Tone Answer Back (TAB, Feature 082-1) allows the selection of three different tones or sequences of tones to acknowledge receipt of data or to give "yes-no" responses to requests. One TAB option is required to service two lines and each additional pair of lines requires another TAB option.
2. Voice Answer Back (VAB, Feature 082-2) is required to control the Type 285-8C, -8F, or -8J Audio Unit. One VAB option is required for each pair of lines which are added, as well as for the first two lines.
3. An Audio Unit (Type 285-8C, -8F, or -8J) is needed for audio response capability. The Types 285-8C, -8F, or -8J are used to vary the type of response service given; each can service up to six lines. The Types 285-8D, -8G, and -8K expand the numbers of lines serviced by the Types 285-8C, -8F, and -8J, respectively. One of these expansion options is required for each pair of lines added after the first six.
4. Voice Cylinders (Features 083-1, -2, -3, -4, -5) are chosen according to the audio service required. The accompanying table shows the various combinations of responses possible and the corresponding combinations of audio units and voice cylinders that are required.

EXPANSION OF AUDIO RESPONSE VOCABULARY

Audio Unit	Voice Cylinder	Words Only	Vocabulary Phrases Only	Words and Phrases
285-8C	083-1		31 Phrases	
	083-2	31 Words		Words + Phrases = 31
285-8F	083-3		63 Phrases	
	083-4	63 Words		Words + Phrases = 63
285-8J	083-5	189 Words		

Communication Software

To complement the capability of each type of Series 200 communication control, Honeywell provides appropriate software packages. For example, Honeywell offers two versatile packages of macro routines, collectively known as Communications I/O C, for the Types 286-1, -2 and -3 controls and Message-Mode Communications I/O C, for the Types 286-4 and -5. These software packages perform control functions to govern the flow of input/output data for a wide variety of computer-controlled communication networks.

As with other Series 200 peripheral input/output packages, the user incorporates the I/O macro routines into his data processing program. With Communications I/O C and Message-Mode Communications I/O C, however, the macro routines address remote terminal devices in a communication network, and the flow of data is controlled in real time. The communication network may be composed of teletypewriters, DATASPEED equipment, the IBM 1050, Honeywell Data Stations, and Honeywell VIP devices. Such devices may be addressed individually, or the polling and broadcasting techniques of the package may be applied.

Both Communications I/O C and Message-Mode Communications I/O C initialize the system, process communication and noncommunication device interrupts, send data to and receive data from the remote terminals, translate data from line-transmission codes (5- through 8-level) to Honeywell standard code (and vice versa), arrange line-to-storage connections between the remote stations and the central processor, and detect and notify the user of any error occurrences. A communication program using Communications I/O C or Message-Mode Communications I/O C may be executed as an independent operation, or it may be processed as the foreground program with simultaneous background program operation governed by the Interrupt Control D routine.

Message-Mode Communications I/O C is significantly faster than Communications I/O C; however, the operating and programming characteristics of the two packages are similar. Programs written for Communications I/O C can be modified easily for Message-Mode Communications I/O C. The message-mode package enables continuous message transfer between the control and the central processor, with an interrupt occurring only after a message block has been stored in or retrieved from main memory. Communications I/O C, however, transfers data character-by-character, with an interrupt occurring for each character stored in or retrieved from main memory. By transmitting characters in blocks and using advanced buffering techniques, the message-mode package can achieve speeds several times faster than Communications I/O C.

The flexibility inherent in the macro structure of both Communications I/O C and Message-Mode Communications I/O C enables the user to tailor his communication processing program to perform data collection, inquiry handling, data distribution, and message-switching operations for many applications such as billing, inventory control, centralized ordering, and management information services.

PRIMARY ADVANTAGES

- Real-time communication programs can be fashioned to each application at lowest cost, for only those macro routines required for a particular application need be specified.
- Routines operate in minimum computer time due to efficient utilization of high-speed storage, resulting in constantly high throughput rates. (Typical programs utilizing Communications I/O C have achieved a throughput rate of 800 characters per second with a Model 200, increasing to 1600 with a Model 2200; with Message-Mode Communications I/O C, the throughput rate is 7,000 characters per second for the Model 200.)
- No detailed knowledge of the Series 200 communication interface is required for programming, as the macro coding services the interface equipment.
- Routines for both packages are modular, facilitating modification of the user's program and system. For example, additional remote terminals can be included in the network with little effort.
- Data translation to and from Honeywell code is provided for line-transmission codes of applicable remote terminals, thereby expediting the user's processing of transferred data.
- Minor editing of input/output data is available through the use of certain macro routines, allowing the user to perform rapid pre- or post-edit operations.
- Error detection and notification are provided for both line and system conditions, serving to decrease costly delays for error determination.
- Most timing considerations for system data flow are handled by the communications packages, greatly simplifying the user's task.
- Costly checkout time is minimized since I/O and control functions are performed by efficient, thoroughly tested macro routines.

5 Instructions

Series 200 computers possess a repertoire of over 50 powerful instructions that allow the programmer to specify all operations quickly and easily. Each instruction has a standard format which may range in complexity from one simple element — an operation code — to many elements. Each instruction includes an operation code which uniquely identifies the type of operation to be performed. Some instructions also include one or more variant characters which are used either to define an operation in greater detail or to specify literally a piece of data to be used in the operation. Most instructions also include one or two operand addresses.

When an instruction is extracted from memory for execution, the instruction elements are automatically loaded into particular control registers. Specifically, the operand addresses are loaded into the A- and B-address registers, and the variant character, if present, is loaded into the variant register. Thereafter, the contents of these registers are used in controlling the progress of the operation initiated.

Certain instructions can be stated in an abbreviated form to conserve storage space and to shorten instruction execution time. For example, the Extract instruction can be specified in any of the forms:

EXT/A/B
EXT/A
EXT

In executing an abbreviated instruction form, the information left in pertinent control registers from the execution of previous instructions is used in place of the unstated elements which would be included in the standard format of the instruction being executed. For example, in executing the abbreviated form EXT, the values remaining in the A- and B-address registers from the execution of the previous instruction are used as the A- and B-operand addresses for the EXT operation. A series of instructions linked together in this manner by residual register contents is said to be "chained."

It is usually possible to use the abbreviated form, Op Code/A, of instructions whose standard format contains two addresses. The actions resulting from using this form take one of two courses depending upon the instruction. In the execution of some instructions, the contents of the B-address register are used as the unstated B-operand address. In the execution of other instructions, the stated A address is used as the B-operand address.

Instruction Descriptions

Each Series 200 instruction is described on the following pages in terms of four parameters: name, operation code, format, and execution time. The following format is used to describe all instructions.

Instruction Name	Mnemonic Op Code

Description of Instruction Execution

Easycoder Standard and Abbreviated Instruction Formats	Execution Times (microseconds)

For those instructions which cannot be chained, the operations initiated by both standard and allowable abbreviated formats are described. In the case of instructions which can be chained, the operations initiated by abbreviated formats are quite similar to those for the standard formats; therefore, only the operations initiated by the standard format are described. Abbreviated format operations can be inferred from these descriptions. The Op Code/A forms of such instructions are footnoted to indicate whether the stated A address or the value in the B-address register is used as the B-operand address (see above).

The execution times given are not the fastest possible times but are based on realistic situations involving the three-character addressing mode. The data fields referenced by both the A and B addresses are assumed to be five characters long. Execution times are rounded to the nearest whole microsecond. The instruction summary table in Section 6 contains detailed timing information for each instruction.

CONVENTIONS

The following symbols are used in the instruction descriptions to convey the meanings indicated.

SYMBOL	MEANING
A and B	Addresses of main memory locations.
(A) and (B)	The contents of the fields indicated by A and B.
V	Variant character.

The phrases "field indicated by A" and "field at A" are synonymous and refer to the data field whose rightmost limit is location A and whose leftmost limit is marked by the next word-marked character to the left of location A. The expression "instruction indicated by B" refers to the instruction whose operation code is stored in location B.

Fixed-Point Arithmetic Instructions

Eight arithmetic instructions perform both decimal and binary arithmetic: decimal add, decimal subtract, binary add, binary subtract, zero and add, zero and subtract, decimal multiply, and decimal divide. The latter two instructions are not available in the Models 120 and 125.

Decimal arithmetic instructions treat their operands as signed numeric data. Normal algebraic sign control is in effect during the execution of these instructions. Any zone bit configuration other than 10 in the rightmost character of a decimal field causes the field value to be interpreted as positive; 10 indicates a negative field value. In decimal arithmetic instructions when the logical operation is subtraction, if (A) is algebraically larger than (B), a recomplement cycle is performed automatically to convert the result to its true form.

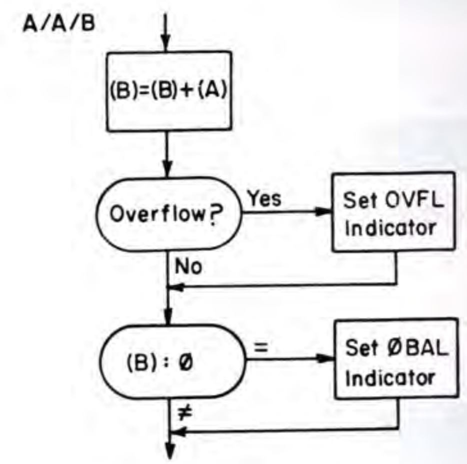
An indicator is set at the completion of each decimal arithmetic operation to indicate the presence or absence of a zero result. A different indicator is set if overflow is sensed. The status of either of these indicators can be tested by a subsequent programmed instruction.

In binary arithmetic operations, the operands are treated as unsigned binary numbers; overflow and zero balance are disregarded.

Decimal Add	A

(A) is added algebraically to (B), and the result is stored in the field at B. B-field zone bits are set to zero except when necessary in the units position to give a true result.

Instruction Format	Execution Time 120	125	200	1200/ 1250	2200	4200
A/A,B	69	58	48	35	25	13
A/A[1]	60	50	42	30	22	13
A	51	43	36	26	19	12



Decimal Add Operation

Decimal Subtract	S

(A) is subtracted algebraically from (B), and the result is stored in the field at B. B-field zone bits are set to zero except when necessary in the units position to give a true result.

Instruction Format	120	125	200	1200/ 1250	2200	4200
S/A,B	69	58	48	35	25	13
S/A[1]	60	50	42	30	22	13
S	51	43	36	26	19	12

Decimal Multiply	M

The signed decimal integer in the field at A is multiplied by the signed decimal integer in the high-order locations of the field at B. The product is stored in

[1] The stated A address is used as the B-operand address.

Instruction Format	120	125	200	1200/1250	2200	4200

the low-order locations of the field at B.

Instruction Format	120	125	200	1200/1250	2200	4200
M/A,B	—	—	419	314	205	82
M/A[2]	—	—	413	310	202	82
M	—	—	407	305	199	81

Decimal Divide — D

The signed decimal integer in the field at A is divided into the signed decimal integer in the field whose leftmost location is B. The quotient and the remainder are stored in the high-order and low-order locations, respectively, of the field at B.

Instruction Format	120	125	200	1200/1250	2200	4200
D/A,B	—	—	219	164	134	56
D/A[2]	—	—	213	160	131	56
D	—	—	207	155	128	55

Binary Add — BA

(A) is added to (B), and the sum is stored in the field at B.

Instruction Format	120	125	200	1200/1250	2200	4200
BA/A,B	69	58	46	35	24	13
BA/A[1]	60	50	40	30	21	13
BA	51	43	34	26	18	12

Binary Subtract — BS

The ones complement of (A) is added to (B), and a simulated carry is added to the sum of the low-order characters. The result is stored in the field at B.

Instruction Format	120	125	200	1200/1250	2200	4200
BS/A,B	69	58	46	35	24	13
BS/A[1]	60	50	40	30	21	13
BS	51	43	34	26	18	12

Zero and Add — ZA

(A) is transferred, right to left, to the field at B. All zone bits in (B), other than those in the sign position (which are normalized), are cleared to zero.

Instruction Format	120	125	200	1200/1250	2200	4200
ZA/A,B	54	45	36	27	19	11
ZA/A[1]	45	38	30	23	16	11
ZA	36	30	24	18	13	10

[1] The stated A address is used as the B-operand address.
[2] The value in the B-address register is used as the B-operand address.
[3] In those timings which vary with the number of bits manipulated, half the maximum number of bits is assumed.

Instruction Format	120	125	200	1200/1250	2200	4200

Zero and Subtract — ZS

(A) is transferred, right to left, to the field at B with the opposite sign. All zone bits in (B), other than those in the sign position, are cleared to zero.

Instruction Format	120	125	200	1200/1250	2200	4200
ZS/A,B	54	45	36	27	19	11
ZS/A[1]	45	38	30	23	16	11
ZS	36	30	24	18	13	10

Floating-Point Arithmetic Instructions[3]

Series 200 scientific processing operations, available in Models 1200, 1250, 2200, and 4200, use four 48-bit accumulators, a low-order register (LOR), and three floating point indicators for exponent overflow, divide check, and multiply overflow. At the most, two of the four accumulators are used in any one operation; these two are designated X and Y for purposes of description.

Many of the scientific processing operations may be performed in either of two forms: (1) accumulator-to-accumulator; and (2) memory-to-accumulator. Accumulator-to-accumulator operations use the contents of the designated floating-point accumulator for the A operand. Memory-to-accumulator operations obtain the A operand from the main memory field specified by the A address stated in the instruction.

Floating Add — AMA or AAA

Instruction Format	120	125	200	1200/1250	2200	4200
AMA/A,XY: (A) is added to (X), and the sum is stored in Y.	—	—	—	38	28	13
AAA/XY: (X) is added to (Y), and the sum is stored in Y.	—	—	—	24	19	9

Floating Subtract — SMA or SAA

Instruction Format	120	125	200	1200/1250	2200	4200
SMA/A,XY: The twos complement of (A) is added to (X), and the result is stored in Y.	—	—	—	38	28	13
SAA/XY: The twos complement of (Y) is added to (X), and the result is stored in Y.	—	—	—	24	19	9

Instruction Format	120	125	200	1200/1250	2200	4200

Floating Multiply — MAM or MAA

Instruction Format	120	125	200	1200/1250	2200	4200
MAM/A,XY: (X) is multiplied by (A). The high-order product is stored in Y; the low-order product is stored in LOR.	—	—	—	50	41	20
MAA/XY: (X) is multiplied by (Y). The high-order product is stored in Y; the low-order product is stored in LOR.	—	—	—	36	33	13

Floating Divide — DMA or DAA

Instruction Format	120	125	200	1200/1250	2200	4200
DMA/A,XY: (A) is divided by (X). The quotient is stored in Y; the remainder is stored in LOR.	—	—	—	56	46	22
DAA/XY: (Y) is divided by (X). The quotient is stored in Y; the remainder is stored in LOR.	—	—	—	41	38	18

Store Floating Accumulator — TAM or TAA

Instruction Format	120	125	200	1200/1250	2200	4200
TAM/A,X–: (X) is stored in A. (X) is not altered.	—	—	—	26	18	7
TAA/XY: (X) is loaded into Y. No normalization occurs.	—	—	—	11	8	5

Load Floating Accumulator — TMA or TAA

Instruction Format	120	125	200	1200/1250	2200	4200
TMA/A,–Y: (A) is loaded into Y. No normalization occurs.	—	—	—	26	18	9
TAA/XY: (X) is loaded into Y. No normalization occurs.	—	—	—	11	8	4

Floating Test and Branch on Accumulator Condition — FBA

Instruction Format	120	125	200	1200/1250	2200	4200
FBA/A,XC: The mantissa portion of (X) is tested for the condition specified by C, the low-order octal digit of the variant. If the condition specified by C is satisfied, program control branches to A. Otherwise, the program continues in sequence.	—	—	—	14	9	5
No Branch						
	—	—	—	15	11	6
Branch						

Branch Conditions for FBA Instruction.

C	CONDITION	C	CONDITION
0	No branch	4	(X) > 0
1	(X) = 0	5	(X) ≧ 0
2	(X) < 0	6	(X) ≠ 0
3	(X) ≦ 0	7	*Unconditional branch

Floating Test and Branch on Indicator — FBI

Instruction Format	120	125	200	1200/1250	2200	4200
FBI/A,OD: The indicators specified by D, the low-order octal digit of the variant, are tested. If any of the indicators are set, control branches to A. Otherwise, the program continues in sequence. All indicators tested are reset.	—	—	—	12	8	4
No Branch						
	—	—	—	14	10	5
Branch						

Branch Conditions for FBI Instruction.

D	INDICATOR	D	INDICATOR
0	No branch	6	Divide check or exponent overflow
1	Multiply overflow		
2	Exponent overflow		
3	Exponent or multiply overflow	7	Divide check, exponent overflow, or multiply overflow
4	Divide check		
5	Divide check or multiply overflow		

Decimal to Binary — DTB

Instruction Format	120	125	200	1200/1250	2200	4200
DTB/A,–Y: The 11-character signed decimal integer whose low-order character is A is converted to a 36-bit binary integer, which is stored in the mantissa portion of Y.	—	—	—	39	30	14

Binary to Decimal — BTD

BTD/A,X–: The mantissa

Instruction Format	120	125	200	1200/ 1250	2200	4200
portion of (X) is converted from a binary integer to a signed decimal integer, which is stored in the 11-character main memory field at A.	—	—	—	41	30	9

Store Low-Order Result — TLM or TLA

Instruction Format	120	125	200	1200/ 1250	2200	4200
TLM/A: (LOR) is stored in A. No normalization occurs.	—	—	—	24	17	7
TLA/–Y: (LOR) is stored in Y. No normalization occurs.	—	—	—	9	7	4

Load Low-Order Result — TML or TAL

Instruction Format	120	125	200	1200/ 1250	2200	4200
TML/A: (A) is loaded into LOR. No normalization occurs.	—	—	—	24	16	9
TAL/X–: (X) is loaded into LOR. No normalization occurs.	—	—	—	9	6	4

Binary Mantissa Shift — BMS

Instruction Format	120	125	200	1200/ 1250	2200	4200
BMS/XM,V: If single precision, the mantissa of (X) is shifted in the mode specified by M, the low-order octal digit of the first variant. If double precision, mantissas of (X) and (LOR) are shifted. The second variant V specifies the number of bits to be shifted.	—	—	—	15	13	8

Binary Integer Multiply — BIM

Instruction Format	120	125	200	1200/ 1250	2200	4200
BIM/A,B: The 4-character fields in memory whose low-order characters are A and B are treated as 24-bit binary integers. The integers are multiplied together; the product is stored in the field specified by B.	—	—	—	45	33	17

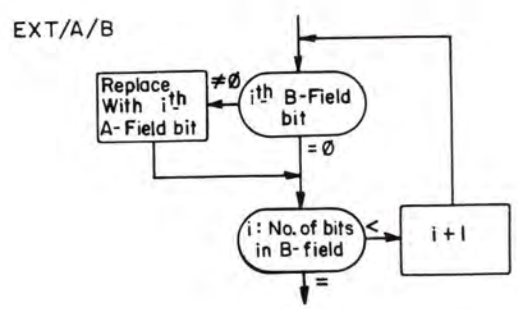
Logic Instructions

Nine instructions are included in this category. Extract, Half Add, and Substitute manipulate data on an individual bit basis, combining bits from two different fields according to rules based on AND and OR logic. Each of the remaining instructions causes a program branch to be performed unconditionally or contingent upon the existence of a precisely defined condition.

Extract — EXT

Each 1-bit in the field at B is replaced with the corresponding bit from (A); all zeros in (B) are undisturbed.

Instruction Format	120	125	200	1200/ 1250	2200	4200
EXT/A,B	69	58	46	35	24	11
EXT/A[1]	60	50	40	30	21	11
EXT	51	43	34	25	18	10



Extract Operation

Half Add — HA

The binary fields at A and B are added without carry, and the result is stored in the field at B.

Instruction Format	120	125	200	1200/ 1250	2200	4200
HA/A,B	69	58	46	35	24	11
HA/A[1]	60	50	40	30	21	11
HA	51	43	34	25	18	10

Substitute — SST

Each bit in the character at B which corresponds to a 1-bit in V is replaced by the corresponding bit in the character at A; other B-bits remain undisturbed.

Instruction Format	120	125	200	1200/ 1250	2200	4200
SST/A,B,V	36	30	24	18	13	8
SST/A,B	33	27	22	17	12	7
SST/A[1]	24	20	16	12	9	7
SST	15	12	10	8	6	6

Compare — C

(B) is compared bit-by-bit with an equal number of characters in the field indicated by A; indicators are set to show the result of the comparison.

Instruction Format	120	125	200	1200/ 1250	2200	4200
C/A,B	57	47	38	29	21	10
C/A[1]	48	40	32	25	18	10
C	39	33	26	21	15	9

Branch — B

The contents of the sequence register are stored in the B-address register, and a program branch to the instruction at A is performed.

Instruction Format	120	125	200	1200/ 1250	2200	4200
B/A	18	15	12	9	8	5

Branch on Condition Test — BCT

If the indicator specified by V is set, the contents of the sequence register are stored in the B-address register and a program branch to the instruction at A is performed; otherwise, program continues in sequence.

Instruction Format	120	125	200	1200/ 1250	2200	4200
BCT/A,V	21	17	14	11	9	5
BCT	9	7	6	5	5	4

Partial list of variant characters for BCT instruction.

VARIANT	INDICATOR	VARIANT	INDICATOR
00	Unconditional		
01	SENSE Sw 1 ON	41	B < A (low compare)
02	SENSE Sw 2 ON	42	B = A
		43	B ≦ A
		44	B > A (high compare)
04	SENSE Sw 3 ON	45	B ≠ A
10	SENSE Sw 4 ON	46	B ≧ A
		50	Overflow
		60	Zero Balance

The conditions in the righthand (or lefthand) column can be combined.

[1] The value in the B-address register is used as the B-operand address.

Branch on Character Condition — BCC

If the character at B contains the type of punctuation and/or sign indicated by V, the contents of the sequence register are stored in the B-address register, and a program branch to the instruction at A is performed; otherwise, the program continues in sequence.

Instruction Format	120	125	200	1200/ 1250	2200	4200
BCC/A,B,V	36	30	24	18	13	7
BCC/A,B	33	28	22	17	12	6
BCC/A[1]	24	20	16	12	9	6
BCC	15	12	10	8	6	5

Partial list of variant characters for BCC instruction.

VARIANT	BRANCH CONDITION
02	B-bit is 1
06	Negative sign
10	Word mark or record mark
12	B-bit is 1 and word mark bit is 1
16	Negative sign and word mark bit is 1
20	Item mark or record mark

Branch if Character Equal — BCE

A program branch to the instruction at A occurs if the character at B is the same as V; otherwise, the program continues in sequence.

Instruction Format	120	125	200	1200/ 1250	2200	4200
BCE/A,B,V	36	30	24	18	13	7
BCE/A,B	33	28	22	17	12	6
BCE/A[1]	24	20	16	12	9	6
BCE	15	12	10	8	6	5

Branch on Bit Equal — BBE

The single character at B is combined bit-by-bit with V according to the rules shown below. This logical product is tested but is not stored in memory. If the result is not equal to zero, the contents of the sequence register are stored in the B-address register, and a program branch to the instruction at A is performed. If the result equals zero, the program continues in sequence.

Instruction Format	120	125	200	1200/ 1250	2200	4200
BBE/A,B,V	36	30	24	18	13	7
BBE/A,B	33	28	22	17	12	6
BBE/A[1]	24	20	16	12	9	6
BBE	15	12	10	8	6	5

Instruction Format — Execution Time: 120 200 1200 2200 4200

Formation of Logical Product in BBE Instruction.

BIT IN B FIELD	BIT IN V	RESULTING BIT
1	1	1
1	0	0
0	1	0
0	0	0

General Control Instructions

The instructions in this category are used to manipulate data within the control memory, to prepare main memory storage areas for the processing of data fields, and to control the sequential selection and interpretation of instructions in the stored program.

Set Word Mark — SW

Word marks are set in the locations specified by A and B; the data in these locations are undisturbed.

SW/A,B	30	25	20	14	11	5
SW/A[1]	21	18	14	11	8	5
SW	12	10	8	6	5	5

Set Item Mark — SI

Item marks are set in A and B; the data in these locations are undisturbed.

SI/A,B	30	25	20	14	11	5
SI/A[1]	21	18	14	11	8	5
SI	12	10	8	6	5	5

Clear Word Mark — CW

Locations A and B are cleared of word marks; the data at these addresses are undisturbed.

CW/A,B	30	25	20	15	11	5
CW/A[1]	21	18	14	11	8	5
CW	12	10	8	6	5	5

Clear Item Mark — CI

Locations A and B are cleared of item marks; the data at these addresses are undisturbed.

CI/A,B	30	25	20	15	11	5
CI/A[1]	21	18	14	11	8	5
CI	12	10	8	6	5	5

[1] The stated A address is used as the B-operand address.

Instruction Format — Execution Time: 120 125 200 1200/1250 2200 4200

Halt — H

H: The machine is halted unconditionally. The sequence register contains the address of the instruction following the halt.

H/A: The contents of the sequence register are transferred to the B-address register, and A is placed in the sequence register; then the machine is halted.

H/A,B: The machine is halted unconditionally. A and B are stored in the address registers as halt identification symbols.

H/A,B,V: The machine is halted unconditionally. A and B are stored in the address registers, and V is stored in the variant register.

H	9	7	6	5	5	5
H/A	18	15	12	9	8	5
H/A,B	27	23	18	14	11	5
H/A,B,V	30	25	20	15	12	6

No Operation — NOP

The contents of the sequence register are incremented; no other operation is performed.

NOP	9	7	6	3	4	4

Change Addressing Mode — CAM

The addresses in all following instructions are interpreted as specified by V until the next CAM instruction is executed. Either or both of the following conditions may be specified by V:

1. Addressing mode (2-, 3-, or 4-character).
2. The "trap" mode of instruction execution. This mode is used to trap operation codes which are not implemented in a given installation and to provide an automatic branch to routines which simulate the nonexistent operation codes.

CAM/V	12	10	8	5	5	4
CAM	9	7	6	3	4	4

Change Sequencing Mode — CSM

CSM: The contents of the sequence register and the change sequence register are interchanged, and the program branches to the address which was previously stored in the change sequence register.

CSM/A: The operations described above are performed. In addition, the A address is loaded into the A-address register.

CSM/A,B: The operations described above are performed. In addition, the B address is loaded into the B-address register.

CSM/A,B,V: The operations described above are performed. In addition, the variant character is loaded into the variant register.

CSM	12	10	8	5	5	4
CSM/A	21	18	14	9	8	5
CSM/A,B	30	25	20	14	11	5
CSM/A,B,V	33	27	22	15	12	5

Store Control Registers — SCR

The value in the control register indicated by V is stored in the field indicated by A. V can be specified to indicate any control register except those used as floating-point accumulators and work registers.

SCR/A,V	30	25	20	15	12	6
SCR/A	27	22	18	14	11	6
SCR	18	15	12	9	8	5

Load Control Registers — LCR

The value in the field indicated by A is moved to the control register indicated by V. V can be specified to indicate any control register except those used as floating-point accumulators and work registers; if the sequence register is specified, a program branch to the instruction at A is performed.

LCR/A,V	30	25	20	15	12	6
LCR/A	27	22	18	14	11	6
LCR	18	15	12	9	8	5

Load Index/Barricade Register — LIB

When basic storage protection is in effect, the character(s) at A (and sometimes A-1) is loaded into the index/barricade register, specifying the number of a 4,096-character portion of memory whose low-order location forms a memory barricade. Data cannot be delivered to memory locations above the barricade unless processing is in the interrupt mode.

When storage protection with base relocation is in effect (format LIB/A,B), the character(s) at B (and sometimes B-1) is loaded into the base relocation register, specifying the number of a 4,096-character portion of memory. The number augments all memory references made in the noninterrupt mode. The index/barricade register is loaded as described above, but the barricade is relocated relative to the base relocation address, so that data cannot be delivered to memory locations above the barricade or below the base relocation address unless processing is in the interrupt mode.

LIB/A,B	—	—	—	18	13	9
LIB/A	—	—	—	11	8	6

Store Index/Barricade Register — SIB

When basic storage protection is in effect, the contents of the index/barricade register are stored in A and sometimes A-1). All bit positions not required to store these contents are cleared to zeros.

When storage protection with base relocation is in effect (format SIB/A,B), the contents of the index/barricade register are stored as above, and the contents of the base relocation register are stored in B (and sometimes B-1). All bit positions not required to store these contents are cleared to zeros.

SIB/A,B	—	—	—	18	13	8
SIB/A	—	—	—	11	8	6

Interrupt Control Instructions

The normal processing sequence can be interrupted by any one of four sources: (1) a peripheral control; (2) the operator's control panel or console; (3) an internal processor condition which violates a protected portion of memory; or (4) a programmed instruction. Program control automatically branches to a stored routine which identifies the source of interruption and services the condition which caused the interruption.

Four instructions are used in conjunction with the automatic interrupt facility. The Monitor Call instruction is one of the four interrupt sources. The remaining three instructions, coded in the interrupt routine, help identify the source, store information which will be needed when the normal processing sequence is resumed, restore this information, and cause the return to the interrupted program.

Monitor Call	MC

The processor is interrupted as follows: the source indicators are set to show that the Monitor Call instruction is the source of interruption, the settings of pertinent processor indicators are stored, the arithmetic indicators are cleared, the contents of the sequence register and the external interrupt register are interchanged, the program branches to the address previously stored in the external interrupt register, and the processor enters the 3-character, non-trap, external interrupt mode of operation.

Instruction Format	120	125	200	1200/1250	2200	4200
MC	9	7	6	3	4	4

Store Variant and Indicators	SVI

Up to six characters of indicator and register status information, as specified by V, are stored in consecutive locations following V (see accompanying table).

Instruction Format	120	125	200	1200/1250	2200	4200
SVI/V	30	25	18	17	11	8

Information Stored by SVI Instruction.

V BIT	CORRESPONDING REGISTERS AND INDICATORS
0	Variant register
1	Arithmetic, comparison, address mode, and trap mode indicators
2	Auxiliary indicators register
3	Scientific unit indicators; interrupt masks
4	Storage protection indicators; instruction timeout allow and S-mode indicators; in some cases, the internal interrupt mode indicator
5	Interrupt source indicators; instruction timeout indicator

Restore Variant and Indicators	RVI

Up to five consecutive characters starting at A are loaded into the processor control indicators and registers specified by V. These characters were stored previously by an SVI instruction. The correspondence between V bit positions and the registers and indicators whose previous contents are being restored is the same as for the SVI instruction, except that bit 5 of V is not used by RVI.

Instruction Format	120	125	200	1200/1250	2200	4200
RVI/A,V	36	30	22	17	13	7

Resume Normal Mode	RNM

The contents of the pertinent interrupt register and the sequence register are interchanged, and program

control branches from the interrupt routine being executed to the address contained in the sequence register. Pertinent control information, previously stored when interruption occurred, is restored, and A and B (if stated) are stored in the respective address registers.

Instruction Format	120	125	200	1200/1250	2200	4200
RNM/A,B	30	25	20	14	12	5
RNM/A[1]	21	18	14	9	9	5
RNM	12	10	8	5	6	4

Data Move Instructions

These instructions are used to move data within the main memory. Control over the placement of punctuation can be exercised, and data can be translated to other forms as it is moved.

Move Characters to Word Mark	MCW

Data and item marks are moved from the field indicated by A to the field indicated by B.

Instruction Format	120	125	200	1200/1250	2200	4200
MCW/A,B	54	45	36	27	19	9
MCW/A[1]	45	37	30	23	16	9
MCW	36	30	24	18	13	8

Load Characters to A-Field Word Mark	LCA

The data and punctuation in the field indicated by A are moved to the field indicated by B.

Instruction Format	120	125	200	1200/1250	2200	4200
LCA/A,B	54	45	36	27	19	8
LCA/A[1]	45	37	30	23	16	8
LCA	36	30	24	18	13	8

Extended Move	EXM

A specified amount of the data and/or the punctuation in the field indicated by A is moved to the field indicated by B. Different variant characters can be specified to control:

1. whether or not punctuation marks are moved;
2. the type(s) of punctuation moved;
3. whether or not data is moved;
4. the direction in which the source field is to be scanned; and
5. how much of the source field is to be scanned.

[1] The value in the B-address register is used as the B-operand address.

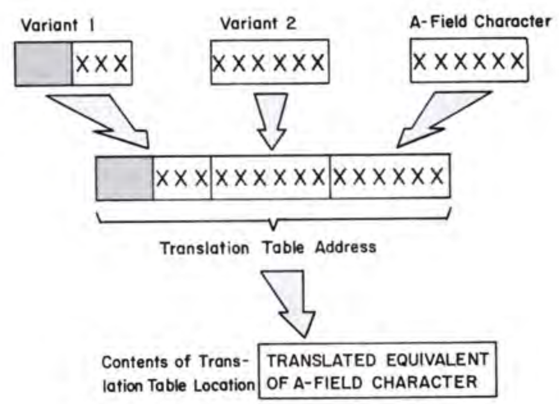
Instruction Format	120	125	200	1200/1250	2200	4200
EXM/A,B,V	57	47	38	29	20	9
EXM/A,B	54	45	36	27	19	8
EXM/A[1]	45	37	30	23	16	8
EXM	36	30	24	18	13	8

Move and Translate	MAT

The characters in the field indicated by A are successively translated according to the contents of a stored translation table; their equivalents are placed in corresponding successive locations in the field indicated by B. The translation table address (binary) of an equivalent character is formed by placing together the binary equivalents of V_1, V_2, and the character to be translated, in sequence.

This instruction and MIT, below, find particular application in the solution of code conversion problems. For example, it can be used to convert easily and efficiently between central processor codes and peripheral media codes having different configurations.

Instruction Format	120	125	200	1200/1250	2200	4200
MAT/A,B,V_1V_2	72	60	48	36	24	13



Formation of Translation Table Address of Equivalent

Move Item and Translate	MIT

The code characters in the item indicated by A are successively translated according to the contents of a stored translation table; their equivalents are placed

[1] The value in the B-address register is used as the B-operand address.

[2] The execution times presented here are based on the assumption that five moves are made. For scan times, refer to the instruction summary table in Section 6.

[3] The execution times presented here are based on the assumption of a five-character search argument and a five-field table.

in the item indicated by B. Characters to be translated and their equivalents may be up to 12 bits in length; the size of these characters (i.e., whether contained in one or two six-bit character locations) is specified by V_3. The translation table address of an equivalent character is formed by combining V_1, V_2, and the character to be translated in a manner similar to MIT, above.

Instruction Format	120	125	200	1200/1250	2200	4200
MIT/A,B,V_1,V_2,V_3	75	63	50	38	25	14

Move or Scan[2]	MOS

Data is moved from left to right or from right to left serially by character, from the field indicated by A to the field indicated by B, under control of V. Characters may be moved in whole or in part, or merely scanned.

Instruction Format	120	125	200	1200/1250	2200	4200
MOS/A,B,V	–	–	–	36	25	17
MOS/A,B	–	–	–	35	24	16
MOS/A	–	–	–	30	21	16
MOS	–	–	–	26	18	15

Table Lookup[3]	TLU

A specifies the location of a search argument; B specifies the location of the rightmost field in a table. V specifies a functional relationship between the search argument and a table argument, satisfaction of which will terminate the search operation and set the appropriate comparison indicators.

Instruction Format	120	125	200	1200/1250	2200	4200
TLU/A,B,V	–	–	–	59	40	29
TLU/A,B	–	–	–	57	39	29
TLU/A	–	–	–	53	36	29
TLU	–	–	–	48	33	28

Edit Instruction

Move Characters and Edit	MCE

The contents of the field indicated by A are moved into replaceable character positions in the edit control word in the field indicated by B; then the contents of the B field are edited to suppress unneeded credit and minus symbols, zeros, and commas, and to insert blanks, asterisks, and dollar signs where necessary.

Instruction Format	120	125	200	1200/1250	2200	4200
MCE/A,B	129	108	66	65	44	14
MCE/A[1]	120	100	60	60	41	14
MCE	111	93	54	56	38	13

Input/Output Instructions

Effective control over data transfers between the central processor and peripheral units and over the peripheral units themselves is maintained by the use of two basic instructions: Peripheral Data Transfer (PDT) and Peripheral Control and Branch (PCB). The PDT instruction is used to initiate data transfer operations and certain other related operations, such as backspace magnetic tape and erase magnetic tape.

The PCB instruction can perform four distinct functions: (1) cause a program branch to be performed contingent upon the setting of peripheral condition indicators; (2) change the operational mode of a peripheral control; (3) initiate strictly mechanical (non-data transfer) operations; and (4) allow a peripheral control to interrupt (or direct it not to interrupt) the central processor when data transfer is completed.

Peripheral Data Transfer — PDT

PDT/A, C_1, C_2, ... , C_n: Data is transferred between the field indicated by A and the control unit and peripheral device indicated by control characters C_2 and C_3, using the read/write channel(s) indicated by C_1. The number of control characters (n in the instruction format) varies with the device being addressed.

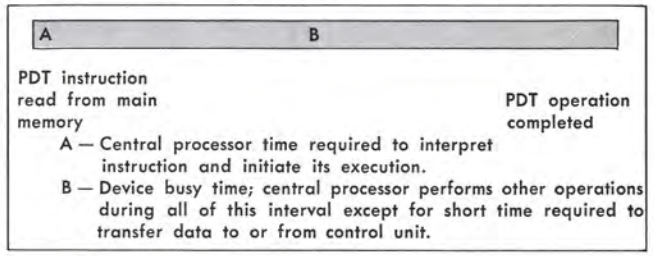
A write operation is terminated when all of the data in the output field has been transferred. A read operation is terminated when the memory input area has been filled or when a standard unit of information, such as a record or a card, has been read.

The timing of a PDT instruction involves three considerations:

1. the time required by the central processor to interpret the instruction (eight memory cycles average time);
2. the time required by the processor to transfer data to or from the peripheral control involved (approximately one memory cycle per character); and
3. the amount of time the peripheral device is busy (varies according to the device involved).

The amount of central processor time required during the entire peripheral operation is relatively small (from .001% to approximately 30%, depending on the processor and peripheral device involved).

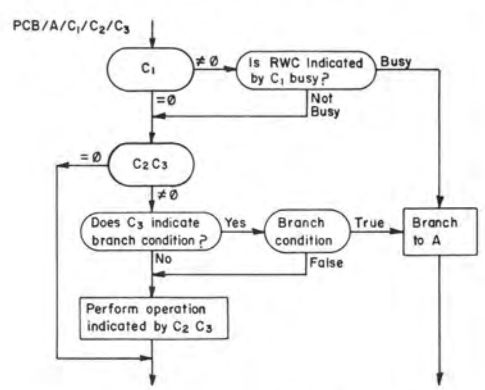
TIME RELATIONSHIPS IN EXECUTION OF PDT INSTRUCTION



Peripheral Control and Branch — PCB

PCB/A,C_1, . . .,C_n: If C_1 specifies a read/write channel, the program branches to the instruction at A if the channel is busy. If the channel is not busy, or if no channel is specified, the operation(s) specified by control characters following C_1 are performed. These operations are summarized below.

1. Test and branch operations — PCB instruction configurations are available to test for conditions such as peripheral control busy, error in last card punched, loss-of-transmission error detected by communication control, etc. If the condition exists, a program branch to the instruction at A is performed.
2. Mode change operations — The PCB instruction can direct the central processor to condition a specified peripheral control for operation in a specific mode. For instance, the card reader control can be conditioned to read subsequent cards in Hollerith mode or in direct transcription mode, depending on the control characters of the PCB instruction. That control could also be directed to reject cards which contain illegal punches, to generate a busy signal if illegally punched cards are read, or both, depending on the particular control characters specified.



Peripheral Control and Branch Operation

3. Single-occurrence peripheral device operations — The PCB instruction can be used to direct the single performance of a non-data-transfer peripheral operation (for example, rewind magnetic tape).
4. Peripheral interrupt operations — Any Series 200 peripheral control can be allowed to interrupt (or directed not to interrupt) the central processor. If allowed, the interrupt signal is generated automatically at the conclusion of a data transfer operation.

Central processor time required for executing a PCB instruction varies with the length of the particular instruction and the processor under discussion. Instruction length varies in turn with the type of operation specified, with A-address length, and with whether or not performance of the instruction includes a program branch. In any case, very little central processor time is required (approximately eight memory cycles).

6 Programming Systems

Series 200 hardware reflects extremely advanced technology within the computer industry. To complement the hardware, Honeywell has developed a comprehensive array of software that capitalizes on its processing capabilities. The software is supplied in several versions tailored to fit various equipment configurations and operating environments. The equipment configurations may cover a range of many different memory capacities and peripheral device combinations.

A distinctive feature of Series 200 programming aids is that they are operationally compatible with each other and with the object programs that they produce. Object programs produced by a variety of program preparation aids as well as programs from the software library may all be intermixed on run tapes and processed together.

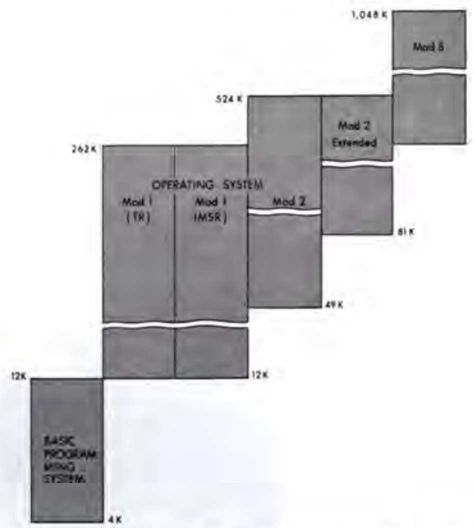
Program compatibility is a built-in feature of Series 200 software. A single machine language is used with all of the Models 120/125/200/1200/1250/2200/4200, allowing the user to run on the Model 4200 programs written for the smaller machines. Thus, software and software-produced object programs which run on even the smallest processor can also run on the larger configurations, and with considerable gain in performance due to the faster cycle times and increased input/output simultaneity of the larger processors.

The Series 200 software available now and planned for the future is grouped into two general categories:

Series 200/Programming Systems – Software which performs computer management functions such as language processing, program checkout and maintenance, job control, I/O control, data editing and transcription, and mathematical processing.

Series 200/Industry Application Systems – Special-purpose software which performs jobs closely related to the functions of the user's organization (e.g., inventory control and sales forecasting). These systems are described in Section 7.

Series 200/Programming Systems are of two types:



Honeywell Series 200 Programming Systems

the Series 200/Basic Programming System, consisting of self-loading, unit-record programs for the main memory range of 4,096 to 12,288 characters; and the Series 200/Operating System. The Series 200/Operating System is divided into five models:

Series 200/Operating System – Mod 1 (Tape Resident)
- Applies to tape-oriented systems in the 12,288 to 262,144 character range;

Series 200/Operating System – Mod 1 (Mass Storage Resident)
- Applies to mass-storage-oriented systems in the 12,288 to 262,144 character range;

Series 200/Operating System – Mod 2
- Applies generally to the 49,192 to 524,288 character range;

Series 200/Operating System – Mod 2 (Extended)
- Applies generally to the 81,920 to 524,288 character range;

Series 200/Operating System – Mod 8
- Applies to Model 8200 systems (262,144 to 1,048,576 characters).

(Refer to *Honeywell Series 200/Model 8200 Summary Description*, Order No. 191, for a discussion of Mod 8.)

System Descriptions

The Basic Programming System and the Operating System models (excluding the Mod 8) are summarized in the following pages. Within the Basic Programming System and the Tape-Resident Mod 1 Operating System, some components illustrate distinctive facets of Honeywell software and have therefore been described at some length.

The degree of integration attained in the Mass-Storage-Resident Mod 1 Operating System, the Mod 2 Operating System, and the Mod 2 (Extended) Operating System precludes the singling out of specific components for expanded discussions. The comprehensiveness of these three systems, however, does not mean that they lack the modularity that is basic to the design of all Honeywell software. A user can make a transition to these systems as his needs demand. Remember that each operating system model offers unique capabilities which reflect the needs of users at various levels of system development. It is the individual user's path of development that determines his choice of operating system components.

Series 200/Basic Programming System

The Series 200/Basic Programming System is designed for card-oriented Series 200 installations which assemble and execute programs individually. Each program in the Series 200/Basic Programming System is an independent entity consisting of facilities for self-loading, data manipulation/specialization, and diagnostic analysis. The prime requirements of an "open shop" installation, where jobs are scheduled and executed on a demand basis, are flexibility and simplicity of operation. These requirements are reflected in the design features of the Basic Programming System, enabling a self-loading, unit-record type of operation for Series 200 processors having main memory capacities in the approximate range of 4,096 to 12,288 characters.

The associated peripheral array for the Series 200/Basic Programming System need include only a card reader and a card punch, although most installations will also have use for a high-speed printer. Magnetic tape drives may be added to this array, providing a more efficient storage medium. Alternatively, program storage and data input/output functions can be allocated to paper tape equipment. Finally, the Basic Programming System includes provisions for storage of frequently used systems or production programs on a self-loading tape (SLT) to increase the efficiency of a card-oriented environment.

OUTSTANDING COMPONENTS

The Basic Programming System is highlighted by four unique elements—COBOL Compiler B, the Easytab system, Critical Path Method A, and the Liberator conversion programs. The accompanying table presents a complete listing of the components in the Basic Programming System.

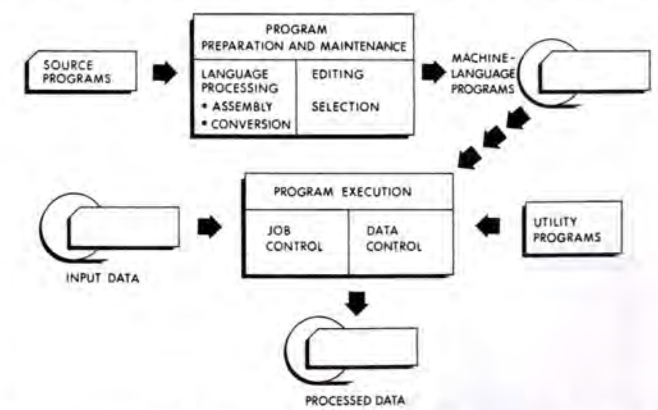
COBOL COMPILER B

The COBOL B language is a simple programming tool which employs commonly used English-language business terms. COBOL Compiler B translates the programmer's COBOL source statements into a machine-language program, i.e., a program intelligible to and executable by the computer.

Honeywell's COBOL compilers are noted for their compilation speed and the efficiency of the resulting machine-language programs. COBOL B requires only 8,192 characters of memory for compilation, the smallest amount of memory yet required for COBOL, but it provides all the language elements necessary to express virtually any business-oriented data processing function.

EASYTAB

The Easytab system is an effective transition tool for tabulating equipment users who are moving up to computers. It consists of three elements: a set of seven preprogrammed utility routines for the performance of common tab functions, COBOL B for those jobs which do not lend themselves to performance by



Functions and Data Flow in the Basic Programming System

the utility routines, and two systems function packages. Easytab retains tab procedures but substitutes the speed and processing capabilities of Series 200 computers for the slower, less efficient operation of tab equipment. The basic processing unit is still an 80-character item, thus retaining much of the basic structure of the tab user's present data processing operations, precluding costly restructuring of data files, and facilitating the orientation of tab installation personnel to the computer environment.

Easytab includes seven preprogrammed (in COBOL) utility routines which handle the bulk of the tab user's workload. These routines are ready to use as soon as the user receives them; he need only compile the routines (a one-time operation), insert parameter cards to specialize the routines to his applications – and the programs are ready to run. Only one or a few parameter cards are required to specialize the utility routines to a given application.

The functions performed by the Easytab utility routines are sorting, merging, selecting, altering, totaling, reproducing, and performing basic input/output functions (for which there is no tab equivalent).

When combined with one utility routine (Sort B), the two systems function packages – Data File Update B and Report Writer B – enable the user to implement completely any basic EDP application with little design or programming effort.

The minimum equipment configuration for the Easytab system is 8,192 characters of main memory, three tape drives, and the advanced programming instructions.

CRITICAL PATH METHOD A

The CPM A program assists the project manager in estimating and controlling the time and technical performance required to gain project objectives. The first step in using CPM is to establish the project's major events and supporting activities and to develop a network. Then, a time estimate specifying the expected duration is applied to each activity. The program then sums the activity time estimates over the network to give the total expected elapsed time for every forward path through the network. The expected elapsed time for the end event of the project is of special importance: the longest expected time path from the beginning event to this event is the *critical path*. Any slippage in an activity on the critical path delays the whole project.

Using the CPM output reports, the project manager can identify critical and noncritical activities and can expedite activities selectively, never expediting any but critical activities and avoiding across-the-board overtime and premium costs.

CPM is particularly useful in planning projects where time estimates for each activity can be made accurately because of abundant previous experience, e.g., building construction, automobile manufacture.

LIBERATOR

Liberator conversion programs embody a uniquely effective approach to competitive language translation which allows users of IBM 1400-series systems to harness the superior throughput and cost/performance characteristics of Series 200 processors without costly reprogramming. Program translation can be performed on either the symbolic or the machine-language level. Additional functions such as input/output operations can also be performed, including appropriate conversions and/or substitutions for competitive routines, thus producing converted programs with optimum operational characteristics on Series 200 processors.

SUMMARY OF ADVANTAGES

- Flexibility and simplicity
- Self-loading, unit-record operation
- Increased efficiency in card environment
- Choice of storage media
- Superior throughput and cost/performance characteristics
- Competitive language translation
- Ease of maintaining and updating program files
- Optimal use of available memory
- Macro routines specialized to user's requirements

COMPONENTS OF BASIC PROGRAMMING SYSTEM

PROGRAM PREPARATION AND MAINTENANCE	UTILITY PROGRAMS
Easycoder Assembler A	½-Inch Tape Handling Routine A
Easycoder Assembler A(P)	¾-Inch Tape Handling Routine A
Easycoder Assembler B	½-Inch Tape Handling Routine A(P)
Library Processor B	¾-Inch Tape Handling Routine A(P)
COBOL Compiler B	Tape Handling Routine B
Bridge Object Program Translator B	Simultaneous Media Conversion A
Easytran 1401	Link A
Easytran Symbolic Translator B	Data Conversion A
Easytran B (120)	Tabulating Simulator A
Condense A	Tabulating Simulator A(120)
Update A	Tabulating Simulator B
Update A(P)	Tabulating Simulator B(120)
Update B	Tape Sort A
Symbolic Update A(P)	Tape Sort A(120)
Memory Dump A	Tape Sort A(P)
Memory Dump A(2)	Collate A
Memory Dump A(3)	Collate A(P)
Report Generator A(2)	**SCIENTIFIC AND MATHEMATICAL SOFTWARE**
Report Generator B(3)	Floating-point Arithmetic/Comparisons A
JOB CONTROL	Floating-point Arithmetic/Comparisons A(N)
Card Loader A	Exponential A
Paper Tape Loader A	Natural Logarithm A
Tape Loader/Search A	Square Root A
DATA CONTROL	Sine A
½-Inch Tape I/O Translator A	Cosine A
½-Inch Tape I/O A	Arc Tangent A
½-Inch Tape I/O B	Linear Equation Solution A
½-Inch Tape I/O B (M)	Floating-point/Fixed-point Conversion A
¾-Inch Tape I/O B	Integer Multiply/Divide A(2V)
½-Inch Tape and Terminal I/O B	Integer Multiply/Divide A(3V)
Paper Tape Read Routine B	Integer Multiply/Divide A(3)
Console I/O B	Integer Multiply/Divide A(2)
	Critical Path Method A

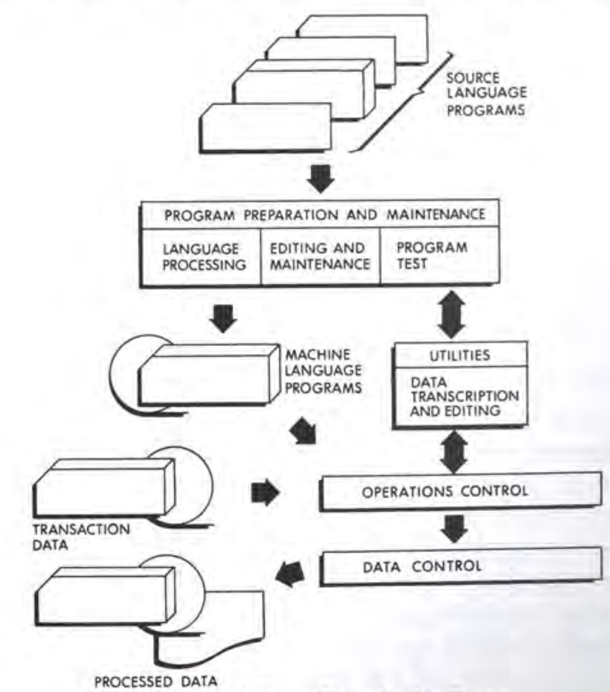
Series 200/Operating System – Mod 1 (Tape Resident)

The Series 200/Operating System – Mod 1 (Tape Resident) is a powerful computer-management system encompassing an extensive set of program preparation and maintenance, job control, data control, and utility functions. These functions can be selectively combined to meet the specific requirements of each user while providing for the systematic exploitation of the equipment capabilities. This building-block approach eliminates uneconomical fixed overhead and guards the user against incurring costs for unnecessary functions and features. It also relieves the user of a host of complex programming and execution supervision tasks. In the Mod 1 Operating System, both hardware and software are modular in structure, compatible in function, and flexible in implementation.

The Tape-Resident Mod 1 Operating System is the unifying element for medium-scale, tape-oriented installations. This system operates on Series 200 computers which have main memory capacities ranging from 12,288 to 262,144 characters; the advanced programming instructions are also required. Peripheral equipment includes from three to six tape drives, a card reader and punch (or a paper tape reader and punch), and a printer. Flexibility at this memory level permits efficient use of mass storage, communication, paper tape, and punched card devices in both independent and semicentralized operations. Basic to the design of Mod 1 are functional program modularity and the standardization of the machine-language formats generated by the various language processors. Standardization of machine-language formats provides the capability to combine the outputs from the various language processors into a single executable job. Modularity is achieved by the structuring of programs into segment units which accomplish specific functions under operating system control. Functional compatibility allows any model of the Series 200 to utilize all of the wide range of peripheral devices and controls in the Series 200 line. Upward compatibility of programs exists for Series 200 computers, restricted only by memory capacity and peripheral equipment.

OUTSTANDING COMPONENTS

The many components of the Tape-Resident Mod 1 Operating System are representative of Honeywell



Functions and Data Flow in the Tape-Resident Mod 1 Operating System leadership in program design. Some of these are discussed individually in the following pages.

EASYCODER ASSEMBLY SYSTEM

Easycoder is designed to provide maximum flexibility and power in programming while retaining simplicity in manipulation. With Easycoder, the user can easily and efficiently prepare a wide variety of programs for operation. The Easycoder language can be used with all Series 200 programming systems. However, it is discussed here only in relation to the Tape-Resident Mod 1 Operating System, which embodies representative implementations of the language.

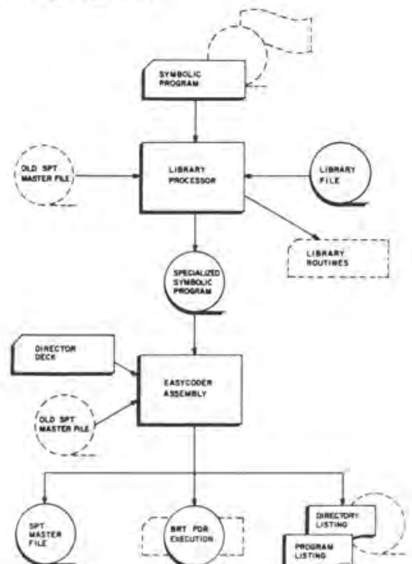
There are two versions of Easycoder for the Tape-Resident Mod 1 Operating System. Each version consists of three elements – a symbolic language, a library processor, and an assembler. Source programs written in the symbolic language are supplemented with precoded symbolic routines by the library processor and assembled into machine-language object programs by the assembler. The Easycoder symbolic language incorporates several types of symbolic statements, using easily remembered mnemonic operation codes and symbolic tags to facilitate actual coding of programs. The library processors permit the user to retrieve routines from a library of already prepared coding by simply issuing calls to the library, thereby eliminating repetitious steps in program preparation. The assemblers, which perform the actual translation from source to machine language, take full advantage of large tape-storage configurations to provide speed in operation. Various file updating functions can also be performed by the assemblers.

The standard version of Easycoder for the Mod 1 environment is Easycoder C. For installations where additional memory capacity and the corresponding hardware features are available, Easycoder D allows the use of floating-point instructions and storage protection. The following information pertains to both versions unless otherwise noted.

Easycoder Symbolic Language

The Easycoder symbolic language incorporates three types of symbolic statements used in writing an Easycoder program – data formatting, assembly control, and data processing. Data formatting statements enable the programmer to reserve and punctuate memory areas and to set up constants without regard for their actual locations in memory. Through the use of assembly control statements, the programmer controls a wide variety of functions related to the process of creating an object program. Data processing statements are the symbolic instructions which are converted by the assembler into the machine-language commands in an object program.

Programming is greatly simplified through the use of macro instructions. These instructions, when specialized by the library processor, cause the insertion of library routines into a source program. One macro call to the library causes a whole routine to be inserted in a symbolic program.



Data Flow in Easycoder Assembly System

Another means for facilitating programming is the use of literals. Binary, decimal, octal, and alphanumeric literals enable the programmer to write in the operands field of a symbolic instruction the actual data (as opposed to the address of the field containing the data) to be operated on by the instruction. Area-defining and address literals may also be employed. An area-defining literal is used to define and reserve a working area in memory without using a separate data formatting statement. An address literal enables the programmer to specify a symbolic address in the operands field of an instruction in such a way that the assembler will use the address as an operand.

Both indexed and indirect addressing can be indicated in Easycoder symbolic language. Symbolic tags of from two to six characters can be used with both versions of Easycoder. In addition, Easycoder D allows the use of ten-character tags for increased flexibility.

Library Processing

The library processors facilitate programming by allowing the programmer to utilize precoded program segments (macro routines) in his program. These macro routines, which are stored in a library file for easy reference, consist of frequently used sequences of instructions in a generalized form. The library processors accept macro calls written in a source program, obtain the macro routines called, specialize them according to parameters submitted by the programmer, and insert them into an Easycodér symbolic program. The library processors can be used to specialize Honeywell-supplied library routines and/or generalized routines written by the user. All levels of nested macro routines can be specialized.

Another function of the library processors is the respecialization of macro routines in an assembled program on a symbolic program tape file. On the subsequent run, Easycoder Assembler D automatically replaces the old macro coding with the respecialized routines generated by the library processor; Easycoder Assembler C requires that the old macro coding be explicitly deleted via the use of DELETE statements.

A third function of the library processors is the punching of symbolic program decks from the library file.

Library Processor C is used prior to the assembly of a source program by Easycoder Assembler C; Library Processor D processes input to Easycoder Assembler D.

Easycoder Assemblers

The Easycoder Assemblers translate programs from symbolic language to machine language and store them on a symbolic program tape (SPT) in both symbolic and binary form. Optionally, an assembled program may be punched on a binary run deck (BRD). An assembly listing and a directory listing are produced. The assembly listing shows the program in source and octal codes; any programming errors are flagged and diagnosed. The directory listing shows the order of all programs on the SPT. Both listings can be recorded on tape for off-line printing if desired. The binary run tape (BRT), another output of the assembler, contains the assembled program in machine-language format only.

There are several advantages to the Easycoder Assembly System. First of all, it shares the compatibility of all Honeywell software components: the language is acceptable to all levels of operating systems. Another aspect of its compatibility is its similarity to Autocoder. This means that the general concepts are familiar to many programmers and that the transition from working with Autocoder to working with the more sophisticated Easycoder is not difficult. Another advantage is its simplicity. Although it allows a programmer to express practically any computation he desires, the language is easy to learn and use.

The extensive set of macro routines that is supplied with Easycoder is another advantage. These routines save the programmer much time and effort. In addition, they eliminate many chances for programming errors. As they are fully proven before being released to a user, they can be called into a program with confidence.

As with all data processing components, one of the main characteristics to be considered is speed. Easycoder assemblers have achieved a very high rating in this category also. For example, a Mod 1 Easycoder Assembler can assemble up to 1000 source statements per minute.

TAPE SORT C

Sorting, the rearranging of randomly ordered input data to produce a specifically ordered data file, is an integral function of most data processing applications. It is characterized by the transfer of large amounts of data into and out of the central processor. These data transfers are limited by the mechanical movement of magnetic tape and thus in many cases do not fully utilize the much faster speeds of the processor. Honeywell's Series 200 systems move data in and out of the processor faster by permitting data to be read by one tape unit and written by another simultaneously. (Depending on the model of central processor used, several other peripheral data transfers may occur at the same time.) Honeywell has also shortened sorting times by reducing the number of times data is moved through the central processor. All Honeywell sort routines are based on the Honeywell-developed read-backward polyphase merge technique, which uses as few as three tape drives while minimizing the number of passes required over the data. This technique can also exploit any number of additional drives (up to six) to achieve even greater sorting efficiency.

An example of Honeywell sort routines is Tape Sort C, a generalized program which can be adapted to different data formats and equipment configurations to handle a wide variety of sorting applications. Tape Sort C can be automatically incorporated into a series of related operations by coding the preceding program to establish the desired sort parameter values before transferring control to the sort process.

Input to Tape Sort C is processed through the program's many operational segments which may be grouped into three logical segments: presort, merge, and last pass. The presort segment accepts the input data in the form of fixed-length units of information called "items" and distributes ordered groups of items called "strings" on from two to five tape reels. The merge segment combines the presorted strings into fewer and longer strings during a series of merge phases, resulting in only one long string on each work tape except one. The last-pass segment further combines these single strings into a contiguous sequence – the sorted file.

The activities performed during the presort and last-pass segments can be augmented by user-written routines, referred to as "own-coding."

COBOL COMPILER D

In the past, many difficulties have been encountered by COBOL users when attempting to integrate COBOL programs into a suitable operating system. Because Honeywell has provided the ability to run COBOL as an integrated part of its various operating systems, there is no other COBOL compiler in operation in a similar environment that begins to approach the operating ease of Honeywell Series 200 COBOL. In addition, the Honeywell COBOL compilers are upward compatible within Series 200.

COBOL Compiler D is a high-speed, low-cost compiler that operates with 16,384 characters of memory. Its speed is exemplified by the fact that it can compile an average program in less than two minutes. Clearly, compiling speed is a major user benefit of COBOL. It is also a major determinant of the cost of program preparation. Studies of compiling speeds and costs by independent organizations have shown that Honeywell's COBOL D costs appreciably less to operate than comparable systems. These studies also verified the fact that Honeywell's COBOL Compiler D is faster than competitive systems and is operable in less main memory.

A major feature of COBOL is that it shifts much of the responsibility for the production of a program from the user's programming staff to the computer manufacturer's COBOL programming staff. In effect, the experience and capabilities of the manufacturer's staff are placed at the disposal of the user via the COBOL compiler. Honeywell's record of achievement in the implementation of COBOL demonstrates the extensive background necessary to create a high-performance product such as COBOL D.

FORTRAN COMPILER D

A Fortran compiler translates programs expressed in a format similar to algebraic equations into machine language. This capability enables scientists, engineers, and managers to apply themselves more fully to a problem without becoming enmeshed in the language of the computer being used.

Honeywell's Fortran Compiler D provides a high-production scientific data processing capability for business installations utilizing Series 200 computers which have from 16,384 to 262,144 characters of main memory. It exploits the magnetic tape capabilities of these installations to provide fast compile and execute speeds. Fortran D features the fullest implementation of Fortran IV available on any machine of comparable size. It provides high throughput rates, rapid turnaround time, the ability to handle large problems, and operating flexibility.

Honeywell's Fortran Compiler D represents a combination of capabilities in many areas of importance to the user. It provides the user with a comprehensive language that enables him to express his procedures efficiently and easily. It performs its translation and code generation quickly and economically and produces programs which perform their tasks efficiently. Above all, it is an integral part of an operating system that is easy to use, yet provides efficient machine performance.

SIMULATION

The use of a computer to simulate real-life situations is growing as business managers recognize the value of this virtually "instant experience." A language called Extended CSL (Control and Simulation Language) enables Series 200 users to simulate business systems and situations. CSL simplifies the modeling of complex industrial and management situations, allowing the most advantageous choice to be made among several possible courses of action.

Any system which deals with chance events is a candidate for simulation. CSL provides simple mnemonic statements in which the basic activities of a system may be described, as well as automatic facilities for generating random numbers and random samples from populations of specified form. The most important feature of CSL is an automatic simulated clock whose time units may represent microseconds, hours, or months, depending on the system being simulated.

In order to simulate any system, the user first identifies the units that will be active in the system and what properties they may possess at different times. The system is then divided into activities and a program sector is written for each activity. An activity may occur according to a schedule determined by the user or it may occur randomly, as in the case of vehicles arriving at a toll booth. In each activity, the objects in the system may change properties. At a ticket counter, for example, a customer may lose the property of being in the waiting line and acquire

COMPONENTS OF TAPE-RESIDENT MOD 1 OPERATING SYSTEM

PROGRAM PREPARATION AND MAINTENANCE
Easycoder Assembler C
Easycoder Assembler D
Library Processor C
Library Processor D
Fortran Compiler D
Fortran Compiler H
COBOL Compiler D
COBOL Compiler H
Easytran Symbolic Translator C
Easytran Symbolic Translator D
Easytran Program Modifier C
Symbolic Program Generator
Program Test Control C
Octal Correction C
Memory Dump C and Tape Dump C
Test Data Generator C
Drum Program Store C
Analyzer C
Update and Select C
Update and Select D
BRT Punch C
SPT Merge C

JOB CONTROL
Tape Loader-Monitor C
Card Loader-Monitor B
Floating Tape Loader-Monitor C
Interrupt Control D
Drum Monitor C
Drum Bootstrap Loader C
List Comments C

DATA CONTROL
½-Inch Tape I/O C
½-Inch Tape and Terminal I/O C
Console I/O C
Standard I/O Calls C
Drum I/O C
Communications I/O C

UTILITY PROGRAMS
Tape Handling Routine C
Simultaneous Media Conversion C
Data Conversion C
Drum Interrogation, Alteration and Loader C
Tape Sort C
Collate C
Tape Sort C(V)
Collate C(V)
Drum Sort C

SCIENTIFIC AND MATHEMATICAL SOFTWARE
Floating-point Arithmetic/Comparisons C
Floating-point Arithmetic/Comparisons C(N)
Exponential C
Natural Logarithm C
Square Root C
Sine C
Cosine C
Arc Tangent C
Linear Equation Solution C
Floating-point/Fixed-point Conversion C
Integer Multiply/Divide C(2V)
Integer Multiply/Divide C(3V)
Integer Multiply/Divide C(3)
Integer Multiply/Divide C(2)
Statistics Package D
Linear Programming Package D
Linear Programming System H
Linear Programming System K[1]
PERT Time C
Control and Simulation Language
Critical Path Method C
Science Library

[1] Requires Feature 1100 or 1101 (Scientific Unit) and 65.536 characters of main memory.

the property of being served. Finally, the user decides how long he wishes the simulated clock to run before automatically completing the job by initiating the printing of the simulation results.

During the simulation, data is stored in arrays and histograms. It may then be analyzed by powerful, Fortran-like arithmetic statements and displayed by means of simple, straightforward output commands. Management can then decide whether the policies simulated are satisfactory or whether others should be tried.

For example, simulation can be used to decide whether or not a new assembly line should be installed in a factory, how many cashiers are needed during peak hours in a supermarket, or even how many calls per hour can be handled by a telephone switching system. To simulate a system requires only that it be possible to divide it into elementary activities and that the system be essentially time-dependent.

The major advantage of simulation is that several alternatives can be simulated at a cost many times less than that of trying out even one alternative in a real-life environment.

PERT TIME C

PERT Time is based on the principle that a small percentage of a project's activities control the schedule for the entire project. Therefore, proper handling of these critical activities – once they are recognized – will facilitate the achievement of project objectives in a more direct and economical manner.

PERT Time C is a program which gives a project manager powerful assistance in planning and scheduling large projects. The advantages of PERT Time C lie in its ability to analyze networks of great complexity (up to 2,000 activities) while operating with only 12,288 characters of main memory. This network analysis enables managers to plan and control activities directed toward the completion of a particular project, to forecast critical areas, and to initiate corrective action if necessary.

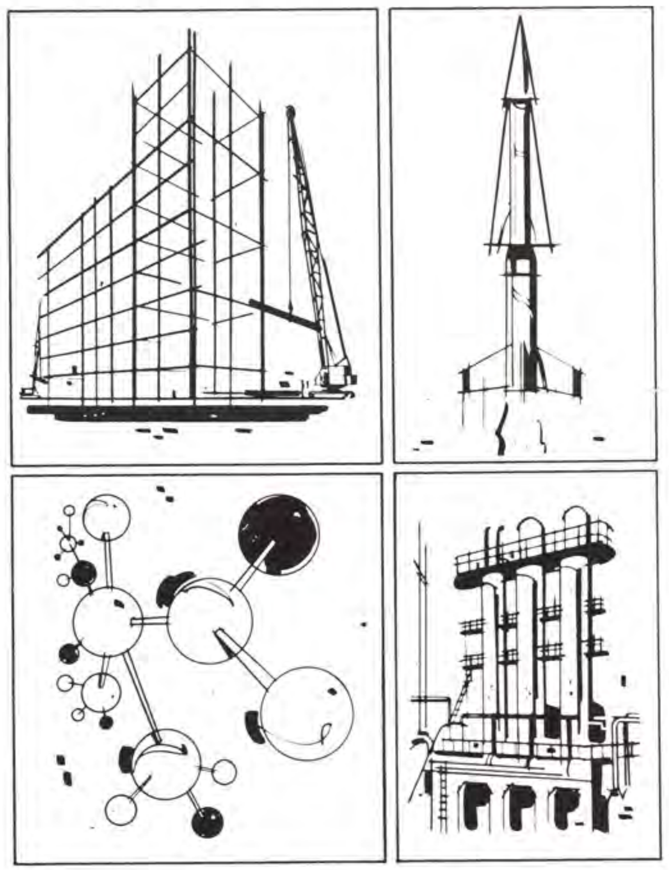
PERT Time C includes many of the capabilities found in larger-scale PERT programs, but it is unique in that it can calculate the number of workers required each workday for five craft categories. The program computes schedules for five-, six-, or seven-day work-weeks. Each activity can be completely defined on one input card. The duration of each activity, and of the entire project, can be more accurately estimated because a flexible range of time estimates may be prepared as input to the program.

PERT Time C is similar to Critical Path Method (CPM) in some respects. Both are designed to assist the project manager in estimating the duration of activities along the most critical path to the desired objective. However, CPM allows only a single time estimate, whereas PERT Time C permits either one or three time estimates for each activity (optimistic, pessimistic, and most likely).

LINEAR PROGRAMMING

Linear programming is a scientific procedure for determining the best design of a certain type of system, given the technology, specifications, and objective of the system. For example, it has proved very useful in the operation of oil refineries and food mixing plants, in the assignment of personnel to various tasks, and the transportation of goods between factories and warehouses. Once the characteristics of a system have been determined, they are translated into a mathematical statement called a model. The optimal solution or best design is then computed using a system such as Linear Programming Package D, Linear Programming System H or Linear Programming System K. The main difference between the systems is the size of the models which can be solved. Each successive system has additional capabilities which allow for greater flexibility of operation and provide for detailed analysis of the proposed model and of changes to the model. The manager, using the best design provided by the computer solution, is then assured of making optimum use of his resources and of meeting his objectives. He has an awareness of the potential trouble spots in his system and is able to react swiftly to changing technology and requirements or to changing objectives.

LP System H is designed to solve problems much larger than those which can be solved by the smaller LP Package D. The upward compatibility between the two systems provides the Series 200 user with important growth potential. Expanding problems (i.e., those problems which have increasing numbers of variables and constraints) which outgrow the capabilities of LP Package D can be transferred to LP System H without any modification of input data. Similarly, LP System K is designed to solve problems larger than those which can be solved by LP System H; expanding problems can be transferred to LP System K from LP System H without any modification of input data.



SCIENCE LIBRARY

The Science Library is a collection of Fortran subroutines, fully supported by Honeywell for use by scientists and engineers in the applied sciences. These subroutines can be incorporated into main programs to solve a broad range of common problems in:

- Statistics
- Polynomial Operations
- General Matrix Operations
- Integration
- Numerical Analysis
- Graph Plotting
- Differential Equations
- Other Mathematical Functions

For busy research scientists and development engineers, the library is an effective, timesaving tool. Rather than spending several days preparing scientific subroutines, they can utilize the tested library subroutines with the assurance that highly reliable solutions will be attained on the first program run.

SUMMARY OF ADVANTAGES

- Relieves the operator of detailed and burdensome execution supervision
- Optimizes throughput by automating operations and minimizing human intervention
- Assures maximum use of the central processor and peripheral units through multiprogramming
- Assures program compatibility as an installation grows
- Enables running of user-written programs, library programs, and systems programs as integral parts of the operating system
- Provides execution of stacked jobs without operator intervention
- Allows tailoring and specialization of precoded, fully tested library routines
- Enables automatic monitoring and documentation of test and production runs
- Provides for segmentation of programs to conserve memory space
- Maintains systems programs simply and economically
- Standardizes operating procedures
- Modular design permits operation with only the needed functions and features

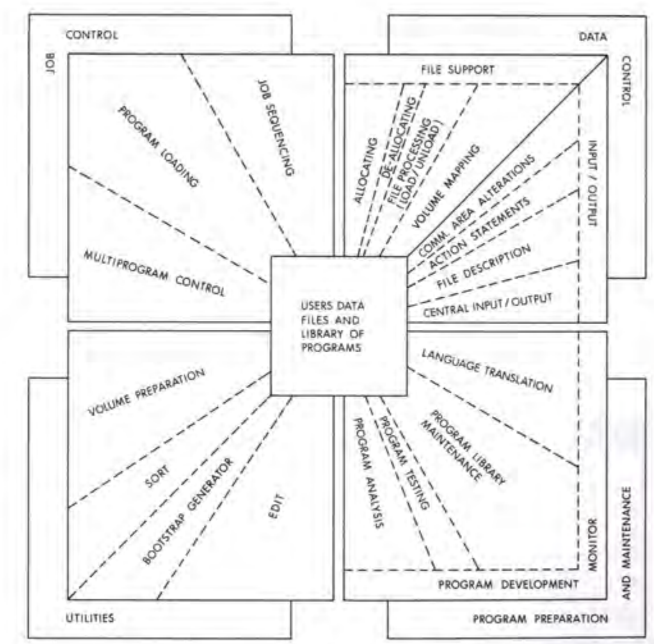
Series 200/Operating System — Mod 1 (Mass Storage Resident)

The Series 200/Operating System – Mod 1 (Mass Storage Resident) is a computer-management system designed for medium-scale installations having a mass storage device and from 12,288 to 262,144 characters of main memory. Incorporated into the system are job control, data control, program preparation and maintenance, and utility functions. Routines are selected and combined to meet each user's specific data processing requirements and to exploit systematically the capabilities of his equipment configuration. Because the system runs with only the necessary routines and equipment, the user is not faced with a costly fixed overhead.

The Mass-Storage-Resident (MSR) Operating System equipment configuration is 12,288 characters of main memory, one mass storage device and control unit, a card reader and a card punch (or a card reader/punch), a high-speed printer, and the advanced programming instructions. However, some programs do not require all of the preceding equipment, while others require more than 12,288 characters of main memory. The system also permits the use of magnetic tape, paper tape, and communication devices. The design of the software modules is such that effective use is made of additional main memory, mass storage, and peripheral equipment when it is available. These modules are linked in a common structure of data management conventions and operating procedures and are controlled by a common supervisor. Each individual module, however, is capable of performing its own task in an independent manner.

OPERATING SYSTEM

The MSR Operating System is a comprehensive package of integrated software modules which are designed to assist the user in making the most efficient use of his mass storage equipment. The four major functions – job control, data control, program preparation and maintenance, and utility – embrace the various routines that are described below.



Functions of the Mass Storage Resident Mod 1 Operating System

JOB CONTROL

All operations in the MSR Operating System are performed under the general control of a supervisor program. Portions of this program are permanently resident in main memory, while other portions are loaded from mass storage as required. A communication area is fixed in lower memory, but the major portion of the supervisor can be "floated" to the highest memory locations available.

The job control function performs job sequencing, program loading, and multiprogram control. Job sequencing effects automatic transition from one job to the next. Program loading encompasses all facets of program segment loading, including locating the desired segment in the mass-storage-resident program file upon request. Multiprogram control allocates the processing time of the central processor in such a way that a peripherally oriented program can be run simultaneously with a data processing program.

DATA CONTROL

Data control involves both conventions and functions. The conventions provide a coherent set of rules and structures which ease the task of managing the total data of an installation. They are designed to provide full control over the user's data, efficiency in processing the data, ease of programming, and ease of operation. The functions are concerned with the control of and access to data files. Two types of functions are included—file support and input/output. The file support functions create, organize, and reorganize files in addition to converting these files from one storage medium to another. The input/output functions retrieve and write data within the files.

File Support

File support creates three types of file organization:

1. Sequential – in which the data is organized for sequential access,
2. Direct access – in which the data is organized for random access, and
3. Indexed sequential – in which the data is organized for both sequential and random access.

File support includes allocate and de-allocate, load and unload, and map routines. The allocate routine reserves an area of a mass storage volume (for example, a disk pack) for a file as described by the user, modifies the volume directory as necessary, and formats and initializes the area if necessary. The de-allocate routine removes a file from a mass storage volume. The load routine loads data into a previously allocated file, establishes the sequence of items, and sets up indexes as required by the file organization. The load routine can accept input from one of several different input media. The unload routine unloads data from a mass storage file, reorganizing the data so that storage utilization and access time are improved when the file is reloaded. The unload routine can produce output on one of several different output media. The map routine produces three types of printed listings that relate the contents of a volume directory.

Input/Output

The input/output functions equip the programmer with macro routines for access to files that are arranged in one of the three supported file organizations. Four types of macro routines – action statements, file description, communications area alteration, and central input/output – are provided. The programmer selects the appropriate macro routines from each type, depending on the access mode desired, the storage devices used, and the number and organization of his files.

COMPONENTS OF MASS-STORAGE-RESIDENT — MOD 1 OPERATING SYSTEM

PROGRAM PREPARATION AND MAINTENANCE	
Mass Storage Easycoder Assembler C (D-level language)	
Library File Update C	
Executable Program File Update C	
Program Testing C	
Easycoder Analyzer C	
COBOL C (B-level language)	**SCIENTIFIC AND MATHEMATICAL SOFTWARE**
COBOL F (D-level language)	Floating-point Arithmetic/Comparisons C
COBOL I (H-level language)	Floating-point Arithmetic/Comparisons C(N)
Fortran F (D-level language)	Exponential C
JOB CONTROL	Natural Logarithm C
Supervisor C	Square Root C
DATA CONTROL	Sine C
Logical I/O C	Cosine C
Physical I/O C	Arc Tangent C
File Support C	Linear Equation Solution C
UTILITY PROGRAMS	Floating-point/Fixed-point Conversion C
Mass Storage Sort C	Integer Multiply/Divide C(2V)
Mass Storage Edit C	Integer Multiply/Divide C(3V)
Volume Preparation C	Integer Multiply/Divide C(3)
Bootstrap Generator C	Integer Multiply/Divide C(2)
	Control and Simulation Language
	Science Library

Action statements name the actions the programmer wants performed and the files to which they are addressed. The file description macro statements describe the structure of the program's files. There must be one file description statement for every file in the program. The communications area alteration macros can interrogate and alter the contents of the file description macro. The central input/output routines perform the common logical input/output operations for all files, control the sharing of physical resources among the peripheral devices, and handle physical input/output operations.

PROGRAM PREPARATION AND MAINTENANCE

Program preparation and maintenance includes routines for language translation, program library maintenance, program analysis, and program testing. A monitor within this function controls the automatic sequencing of the various steps in a program development job.

The language translators include an Easycoder assembler, a COBOL compiler, and a Fortran compiler. The symbolic language of the assembler is comparable to Easycoder D in the Series 200/Operating System – Mod 1 (Tape Resident); library processing is an integral part of the assembler. The COBOL C language is comparable to COBOL B in the Series 200/Basic Programming System, and COBOL F and I are comparable to COBOL D and H, respectively, in the Mod 1 Tape-Resident Operating System. The Fortran language (F) is comparable to Fortran D in the Mod 1 Tape-Resident Operating System.

COBOL F and I both allow the user to have a disk-, drum-, or tape-resident compiler which creates object coding for residency on disk, drum, or tape. All necessary verbs for processing with data files stored on disk are included in these versions. Similarly, Fortran F allows the user to have a disk-resident compiler. It also handles all normal types of disk file organization when processing a disk-resident data base.

Program file maintenance employs routines to maintain either source- or machine-language files on mass storage. The analyzer produces a symbolic cross-referenced listing for an Easycoder symbolic program. Program testing routines allow machine-language patches to be made to a program. Main memory and mass storage dump routines are provided.

UTILITIES

The utility functions perform such common processing tasks as volume preparation, sorting, editing, and bootstrap generation.

The volume preparation routine prepares a mass storage volume for use in the MSR Operating System by checking for bad surface areas, formatting tracks, and establishing the volume label and directory. The mass storage sort operates with one mass storage device. The sorting key may be composed of up to ten separate fields. Key sorting is performed, and other information extracted from the item may be associated with the sorted keys. A mass storage edit routine is provided to edit and print selected areas from mass storage. A bootstrap generator routine creates the mass storage supervisor bootstrap routine on a mass storage volume.

SUMMARY OF ADVANTAGES

- Relieves the operator of detailed and burdensome supervision of execution
- Standardizes operating procedures
- Permits operation with only the needed functions and features
- Enhances the running of user-written programs, library programs, and systems programs as integral parts of the operating system
- Enables programs and files to be located and processed without time-consuming searches
- Makes optimum use of storage surface through a flexible file design that allows the user to specify his record size
- Provides execution of stacked jobs without operator intervention
- Allows tailoring and specialization of precoded, fully tested library routines to meet program needs
- Maintains systems programs simply and economically
- Assures efficient use of the central processor and peripheral units through multiprogramming

Series 200/Operating Systems — Mod 2 and Mod 2 (Extended)

The Series 200/Operating Systems – Mod 2 and Mod 2 (Extended) are both an integrated set of interdependent programs providing the most efficient means for program development and operation. The Mod 2 and Mod 2 (Extended) Operating Systems support the Series 200 Models 1200, 1250, 2200, and 4200 in fulfilling the specific needs of a medium-scale computer installation. Their processing goals include the utilization of proven data techniques and accepted language facilities integrated into a comprehensive system. The operating systems are the interface between the user and the computing system. By reducing the mutual dependence of the user and the computer, the Mod 2 and Mod 2 (Extended) Operating Systems increase throughput, decrease turnaround time, make program preparation and maintenance simpler and more flexible, and standardize operating procedures.

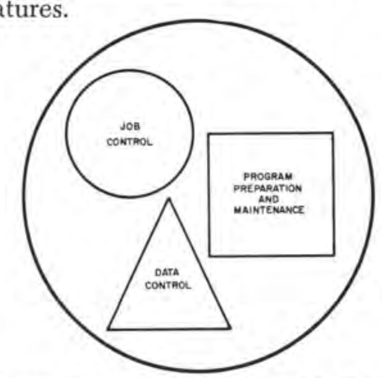
The Mod 2 Operating System is a fully integrated, tape-oriented operating system, enabling stacked-job processing. Language processing includes COBOL, Fortan, and assembly-language processors. System maintenance, centralized I/O control, logical data file handling, relocation and linkage facilities for program modules, and a wide range of utility functions are available.

The Mod 2 (Extended) Operating System is a fully integrated modular operating system functioning in a mixed tape/mass storage environment. This disk-resident operating system contains all the facilities of Mod 2 with additional capabilities enabling multiprogramming and on-line, real-time communications, mass-storage support, and memory and file protection.

The Mod 2 (Extended) Operating System can perform communication processing in a multiprogramming environment, implying the existence of foreground and background programs and the full resident control system (resident monitor, IOFCS, communication supervisor, and foreground scheduler) in core storage. Foreground programs can perform data transcription functions with priority over the background program. Three data transcription tasks can operate concurrently and will convert data from any one peripheral device to any other. These features can exist coincidentally: multiprogramming with both a foreground program operating for on-line, real-time communications (or multi-task data transcription) and a background program processing stacked jobs; or multiprogramming of two stacked-job processing streams.

Mass storage support is provided through advanced data management facilities. Data access methods include sequential, direct access, and indexed sequential. Both system and user programs are stored on disk in executable 8-bit-transfer format (i.e., punctuation bits included), thus enabling the fastest possible loading and unloading of core storage. Core storage protection is assured by support of the program-defined upper and lower memory barricades. Core-storage utilization is optimized by applying the base relocation addressing feature. File protection is governed by software.

The Mod 2 (Extended) Operating System is capable of concurrent dual job processing, enabling two independent job streams to share the same central processing facilities through multiprogramming. Thus, one program can use the input/output facilities while another program can use the central processor's computing facility. This technique of sharing between concurrent programs enables maximum utilization of system features.

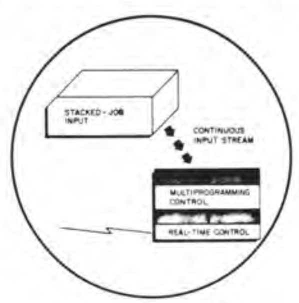


Functions of the Mod 2 and Mod 2 (Extended) Operating Systems

FUNCTIONS OF MOD 2 OPERATING SYSTEMS

JOB CONTROL

Job control under the Mod 2 and Mod 2 (Extended) Operating Systems is handled by two monitors, whose primary function is to provide automatic job-to-job transition. One monitor remains resident in main memory at all times; the other monitor is loaded periodically to assist in the automatic transition process. Most of the human operations involved in transitions between programs within a job and between jobs in the input stack have been absorbed by the Mod 2 operating systems. These clerical duties include collecting the output produced by the previous program, locating the next program to be executed and loading it into memory, and coordinating peripheral device assignments. Job control functions are initiated by system control cards, on which the user schedules program executions and peripheral device assignments. Reducing the idle time between runs by automatic job-to-job transition is one of the ways in which both Mod 2 operating systems increase system throughput.



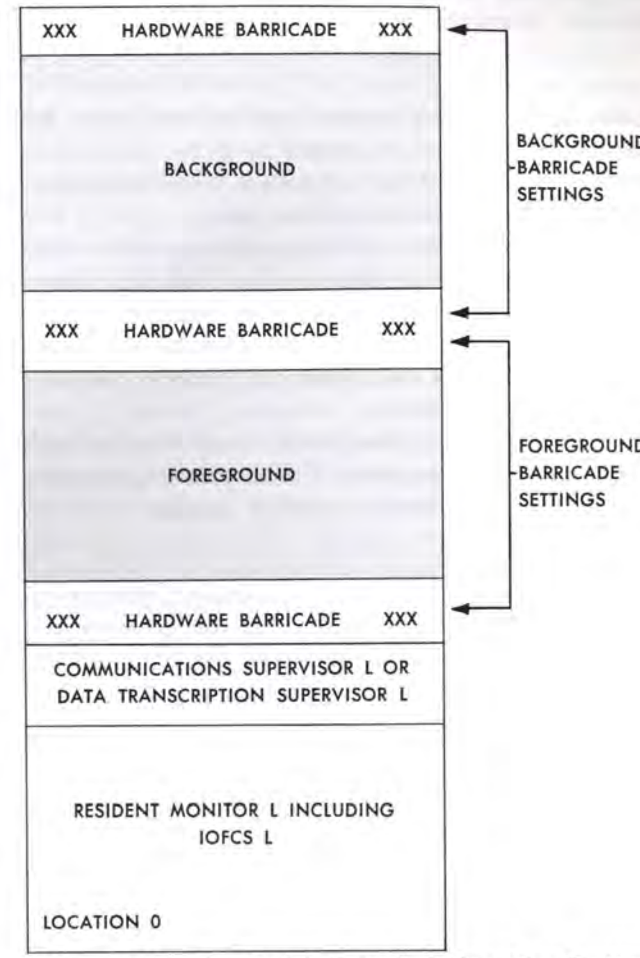
Job Control

Continuous processing of jobs is combined with a method called stacked-job processing, a refinement of the earlier batched-job approach. Under batched-job processing, a single processing function, e.g., compilation, is applied to all jobs in the batch. While a group of jobs may be compiled in succession and then executed in succession, program generation is divorced from the execution of a batch of pregenerated programs. Under stacked-job processing, any number of processing functions such as compilation, maintenance, and execution may be successively applied to the same job. Thus, each job in the input stack is processed to completion before the next job is accepted.

In batched-job processing, the elapsed time between the submission of a job and the receipt of results (turnaround time) is equal to the total processing time for the entire batch which includes the job. In contrast, stacked-job processing dramatically reduces turnaround time for a given job by completing each job before beginning the next.

The job control function performed by the operating systems after loading each program into memory is monitoring, which consists of controlling the internal sequencing of all programs executed under the system. At the proper instant, control is delegated to a program or retrieved from it. Monitoring facilities of the Mod 2 (Extended) Operating System support operation in communication/real-time and multiprogramming environments. Monitoring in a communication environment involves the control of message flow to and from the computer and message processing within the computer. Monitoring in a multiprogramming environment consists of supervising the concurrent execution of two programs. One program is normally peripherally limited and is executed in upper memory. The second program runs in lower memory during the peripheral cycles of the upper-memory program. This optimal utilization of the central processor and peripheral devices is another source of increased throughput under the Mod 2 (Extended) Operating System.

In a real-time multiprogramming environment, job control becomes more involved. Memory is partitioned into foreground and background areas with processing scheduled in each area. Job requests from remote terminals in an on-line, real-time environment must be analyzed; the program requested for execution is scheduled in the foreground area concurrent with execution of the main job stream activity in the background area. When operating in this environment, the foreground is reserved for processing inquiries from remote terminals. A communications supervisor resides permanently in memory with the monitor, as shown below.



Memory Map for Mod 2 (Extended) Operating System

In this environment, programs are called by control cards (background) or by communication requests (foreground).

When a foreground program requires more memory than is available, the background program is temporarily rolled out of memory to create more foreground area. A roll-out/roll-in feature in the resident monitor performs this function.

In lieu of communication activity, data transcription programs can be scheduled in the foreground area. Requests for data transcription activity, submitted by means of console entries, are analyzed, the requested programs are then located in on-line libraries, loaded into the foreground area, and executed under system control. The data transcription supervisor resides in memory during this type of processing in place of the communication supervisor.

Through both hardware and software features, the multiprogramming capability permits execution of more than one program within allotted boundaries and provides protection for each program and the Resident Monitor against destruction.

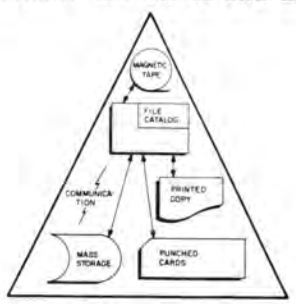
Except for a few input/output devices assigned to certain system files, every I/O device is dedicated to either the lower or the upper program area when the system is generated. Changes in these assignments are made by control cards at system initialization time or through the console between runs.

To obtain efficient multiprogramming with a limited number of I/O devices, the dual job stream capability provides two program modules – Input Reader and Output Writer – to allow a card reader and/or a printer or card punch to be shared between the two job streams.

Another job control function is communication with the operator, advising him of the status of processing and requesting necessary operator actions.

DATA CONTROL

Data control in both Mod 2 operating systems encompasses all functions related to the creation and maintenance of the data base, i.e., the entire collection of information which enters or leaves the computer main memory. The facilities available under data control provide efficient storage, flow, and retrieval of all data in the system. These facilities include two functions: file access and file control.



Data Control

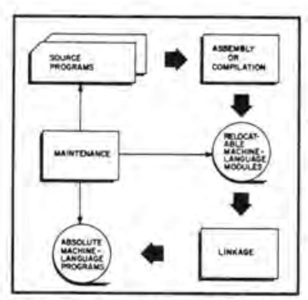
In conjunction with constructing the file catalog, the Mod 2 (Extended) Operating System automatically allocates and partitions space for mass storage files. Storage allocation is complemented by automatic file protection.

The principal file access function is the physical exchange of data between main memory and unit record, magnetic tape, mass storage, and communication equipment. Resident routines execute the data transfer functions, using several different access methods. Other file access functions are associated with data transfer. These other functions are automatic error detection and correction, automatic data buffering, automatic data blocking and unblocking, dynamic scheduling of input/output facilities, and overlapping of processing with input/output operations.

The file control function includes management of logical data files at a level which is independent of the physical characteristics of the files and their storage devices. Files are assigned symbolic names, and the Mod 2 (Extended) Operating System maintains a symbolic file catalog. The file catalog is constructed like a library catalog, so that each file may be indexed, according to its functions, within a set of symbolic classification levels. Upon receiving a request for a file with a given symbolic classification, the Mod 2 (Extended) Operating System consults its symbolic catalog, determines the physical identity and location of the file, and retrieves the file.

PROGRAM PREPARATION AND MAINTENANCE

The most familiar program preparation function is language processing. Programs written in compiler or assembly languages are translated to program modules in relocatable machine language. A program module is the basic program unit in both operating systems. Each module is created independently. Modules are relocatable and can be combined with other modules to fashion a variety of complete programs. The second program preparation function consists of assembling a complete program by selecting specified program modules, providing linkages between the modules, and assigning absolute memory addresses to the relocatable machine code. Complete programs may be built to run anywhere in core storage using any combination of modules. Also, all language processors in both Mod 2 operating systems generate the identical type of relocatable modules. Hence, a complete program may be subdivided into program modules on the basis of physical size, functional breakdown, or the nature of the source language best suited for solving a portion of the total problem.



Program Preparation and Maintenance

The program maintenance functions include adding and deleting modules from the system files and correcting lines within specified modules. Maintenance may be carried out at the source-language, relocatable-code, or absolute-code level. The identical maintenance functions are applied to both system and user programs, and may be used to incorporate user-written modules into the Mod 2 operating systems.

UTILITIES

The automatic debugging facilities of both Mod 2 operating systems include dynamic core and tape dumps. Mod 2 (Extended) offers the data editing and transcription functions which include sorting and merging magnetic tape and mass storage files and performing media transcription operations.

BENEFITS OF THE MOD 2 OPERATING SYSTEMS

EASE OF PROGRAMMING

The relocatable program module is the common denominator of the Mod 2 operating systems. Because they are relocatable, all modules are essentially library routines which the operating system can freely combine. Free communication between program modules is maintained through the standard interface of the operating systems. Because all language processors generate the same basic building blocks (program modules), a programmer is not limited to solving an entire problem in a single source language.

Responsibility for tedious and complex input/output programming is transferred from the user to the centralized routines of the operating systems. Programmers need not be conversant with the unique programming characteristics of any peripheral devices. In addition, both Mod 2 operating systems automatically free the user from allocating buffers, checking file labels, blocking and unblocking records, and error-checking data transfer operations. Also, the operating systems ensure continuous use of the system facilities by maximizing the simultaneity of data flow and internal processing, a capability which is inherent in Series 200 hardware.

Controlling the flow of data to and from peripheral devices is just part of the device independence provided by the Mod 2 operating systems. The operating system also manages the logical data files themselves. The symbolic file catalog within the Mod 2 (Extended) Operating System allows programmers to request data files by using only their symbolic names. The mechanics of locating and retrieving data files are the responsibility of the Mod 2 (Extended) Operating System. The Mod 2 (Extended) Operating System also controls space allocation and formatting on mass storage devices.

Finally, the standardized and automatic debugging facilities of the Mod 2 operating systems, coupled with the brief turnaround time per job, enhance the ease and efficiency of program checkout.

EASE OF OPERATING

A common set of operating procedures is superimposed on both user-written programs and components of both Mod 2 operating systems. Operators do not have to cope with the peculiarities of every program, a fact which simplifies operator training and increases the reliability of machine room operation. In the same fashion, man/machine communication is reduced to a standard dialogue between the operator and the operating systems. Most functions required for automatic job-to-job transition have been programmed into the Mod 2 operating systems.

The Mod 2 (Extended) Operating System also administers internal hardware facilities such as the interrupt system and storage protection, further simplifying the role of the operator. Standardized operating procedures make the mode of operation sensitive to the requirements of each application. Stacked-job processing, batched-job processing, and real-time processing are handled with equal facility.

EASE OF MAINTENANCE AND EXPANSION

Both user programs and Honeywell-supplied components of the Mod 2 and Mod 2 (Extended) Operating Systems are easily modified because of their modular structure. A series of complex, time-consuming programs is required initially to generate some operating systems. However, the same single-phase component of both Mod 2 operating systems which is used to update the system files is also used to create a working version of the operating systems themselves. System generation is both selective and efficient. A personalized operating system is tailored to each installation by incorporating only those system modules required by the user. A typical business-oriented version of the Mod 2 operating systems is generated in less than 15 minutes. System programs and user programs are easily updated without recompiling. For example, additional modules may be added to user programs to take advantage of newly acquired hardware. Additional modules may be added to the operating systems to provide further processing capabilities for growing applications. Also, both operating systems may be expanded by the inclusion of user-written components.

COMPONENTS OF THE MOD 2 AND MOD 2 (EXTENDED) OPERATING SYSTEMS

PROGRAM PREPARATION AND MAINTENANCE	
Mod 2	Mod 2 (Extended)
Assembler J COBOL Compiler J Fortran Compiler J[1] Easytran Symbolic Translator J[2] System Maintenance J Easyauto Symbolic Translator J	Assembler L COBOL Compiler L Fortran Compiler L[1] Easyup Transition L Easycoder L System Maintenance L Easyauto Symbolic Translator L Report Generator L

JOB CONTROL	
Mod 2	Mod 2 (Extended)
Resident Monitor J Linkage Monitor J Transitional Monitor J	Resident Monitor L Linkage Monitor L Transitional Monitor L

DATA CONTROL	
Mod 2	Mod 2 (Extended)
I/O Control System J Data Transcription J	I/O File Control System L Communications Supervisor L Data Transcription L

UTILITY PROGRAMS	
Mod 2	Mod 2 (Extended)
Tape Sort J Utility J	Tape Sort L Mass Storage Sort L Utility L

SCIENTIFIC AND MATHEMATICAL SOFTWARE	
Mod 2	Mod 2 (Extended)
Science Library	Science Library

[1] Requires Feature 1100 or 1101 (Scientific Unit).
[2] Requires six magnetic tapes.

OVERALL BENEFITS

From the perspective of the data processing manager, the convenience and modularity at each level of the Mod 2 and Mod 2 (Extended) Operating Systems are reflected and amplified in the overall efficiency and reliability of the hardware/software complex. Standardized programming and operating procedures provide the most efficient path from initial formulation of a programming problem to final utilization of the solution. Total hardware utilization through multiprogramming and reduced idle time because of automatic job-to-job transition increase throughput under the Mod 2 (Extended) Operating System. At the same time, the stacked-job capability provides comprehensive software service with minimal turnaround time. The flexible framework of the Mod 2 (Extended) Operating System supports growth into applications such as total information and real-time systems. The magnitude and complexity of the functions performed by both Mod 2 operating systems simplify the jobs of programmer, operator, and manager. By furthering the independence of these personnel from the computer, the Mod 2 operating systems allow them to use it more effectively.

MINIMUM HARDWARE REQUIREMENTS

The basic configuration required for the Mod 2 Operating System is a Type 1201, 1251, 2201, or 4201 Central Processor having 49,192 characters of main memory and the Optional Instruction Feature (0191). Also needed are five tape drives, one card reader (or one additional tape drive), one printer (or one additional tape drive), and one operator's console.

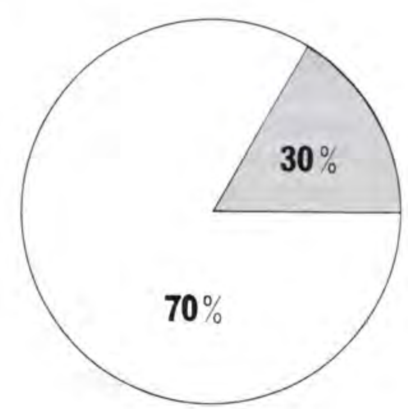
The basic configuration required in order to use the Mod 2 (Extended) Operating System is a Type 1201, 1251, 2201, or 4201 Central Processor equipped with the Extended Multiprogramming and 8-bit Transfer capability (Feature 1120 for the Types 1201 and 1251, Feature 1121 for the Type 2201, and Feature 1118 for the Type 4201). Features 1120 and 1121 require that the Types 1201/1251 and 2201 be equipped with Storage Protect (Features 1114 and 1117, respectively); the Storage Protect capabilities are automatically included in Feature 1118 for the Type 4201. In the case of the Type 1201, 1251, or 2201, the Optional Instruction Package (Feature 0191) is also required. Due to the modularity of the system, memory requirements depend upon the specific version generated by the user. Also needed are one disk device, three tape drives, one card reader (or one additional tape drive), one printer (or one additional tape drive), and one operator's console.

7 Industry Application Systems

To supplement its hardware and computer management software, Honeywell provides its customers with application systems, or "packages," which perform functions peculiar to a specific industry. ARS, for instance, automatically rates shipments for trucking companies; STET is a typesetting program for printers and publishers; CASH does the accounting for distribution companies, and so on. These packages incorporate the systems know-how that Honeywell has acquired through many years of experience in working with customers in a large variety of industries.

Application systems are extremely valuable to a company whether it is taking its first venture into electronic data processing or acquiring a replacement computer to provide greater data processing capability. By applying the appropriate application system, the user can significantly reduce the amount of effort required in planning his initial and subsequent applications, developing an integrated system, and solving a variety of management problems.

A company installing a computer faces a sizable system startup cost. This cost is made up primarily of the salaries of systems and programming personnel needed to analyze, design, and implement the desired computer applications. The system startup cost can be divided into two major elements, as represented in the following diagram.



The major portion of the systems cost is the problem-solving phase; that is, determining in detail:

- the objectives of the job;
- the inputs and outputs of the job, and
- how the job can be performed by a data processing system.

As indicated above, this phase constitutes the largest portion (typically 70 percent) of system startup costs; the remaining 30 percent covers the coding and testing (translating the system into computer language).

Honeywell application packages provide a solution to the greater portion of the system startup problem. Each one consists of an English-language narrative, general and detailed flow charts, item designs, program logic, and input/output formats. In addition, to ease implementation problems, Honeywell provides fully coded and tested programs for applications which are of a general nature and common to the needs of many companies.

The Building-Block Approach

Honeywell recognizes that while it is true that the data processing needs of many companies are similar, they are not identical. Therefore, Honeywell's application packages are composed of "building blocks" at both the system design and programming levels. The building-block approach facilitates individual specialization and tailoring of systems.

The building-block philosophy is also present in Honeywell's over-all approach to application packages for a specific industry. Each package is developed as a subsystem of an integrated management information system. Individual packages are designed to meet the needs of a restricted area of the organization, but with the needs of the whole organization in mind. Thus, the particular package can be used as a separate entity or as part of an integrated system. Once a user has put a basic application into operation, other system functions can be computerized with a minimum of additional information, time, and money.

The following industries are among those served by Honeywell application packages:

- Distribution
- Manufacturing
- Insurance
- Publishing and Printing
- Motor Freight
- Education
- Banking
- Finance
- Data Processing Services
- Retailing

Additional application packages for these and other industries are continually under active analysis and consideration. Following are brief descriptions of Honeywell's available industry application systems.

Publishing and Printing

Honeywell has several systems to offer the newspapers, magazines, and general publishing companies. These include the two STET (Specialized Technique for Efficient Typesetting) systems, handling hyphenation and/or justification for hot-metal or photocomposition typesetters, and an integrated classified advertising system.

STET-1 is a comprehensive typesetting system which provides facilities for the layout, justification, and hyphenation of all types of copy (straight material, classified copy, and tabular material) for hot-metal linecasters. STET-1 is one of a series of linked subroutines written to facilitate modification, extension, and integration into a large system.

A STET monitor is also available which allows a single central processor to do both accounting and typesetting on a shared-time basis.

Photo-STET is a generalized hyphenation and justification system written primarily for photocomposition typesetters, but which is adaptable to other composing devices. Photo-STET includes all the layout facilities of the STET program but has added messages which permit the exploitation of the full flexibilities of any existing photocomposition device. These include disk or grid changes, point-size changes, line spacing, and many others. Care has been taken to ensure compatibility between the STET hot-metal and the Photo-STET (six-level only) input routines; this enables the same operator to keyboard either hot-metal work or photocomposing in a shop using both processes.

All the STET programs have been designed to be readily incorporated into integrated systems. Additional applications are being explored in the areas of classified advertising, book cost estimating, and subscription fulfillment.

Manufacturing

Honeywell is developing Series 200 applications for the manufacturing area within an over-all integrated system called *Factor. Factor* is based on Honeywell's own experience in manufacturing and in helping numerous customers implement systems. Such experience led to the development of *Factor*, whose phased approach enables users to build in subsystem increments as they work toward an integrated management information and control system. Honeywell supplies systems packages and, in some cases, coded packages for selected subsystems to aid the user in implementation.

Factor includes the following subsystems:

- Order Processing
- Forecasting
- Inventory Reporting and Control
- Purchasing
- Manufacturing Engineering File Processing
- Requirements Generation
- Production Scheduling and Control
- Management Planning

FICS – Forecasting for Inventory Control System – is one of the *Factor* subsystems. It uses an advanced exponential smoothing technique in combination with the traditional components of inventory control – economic order quantity, reorder point, and safety stock – to achieve three principal management objectives: (1) to improve customer service, (2) to maintain inventory at lowest possible cost, and (3) to reduce management involvement in routine decision-making.

A simulator (ISIM) is also provided as part of FICS to test the system's accuracy and to predict the effects of proposed changes in policy. ISIM accepts as input live data concerning the user's products and then simulates and analyzes inventory conditions. It can, therefore, be used to test the applicability of FICS to a user's inventory before implementing this system.

TABS – TApe Bill of materials System – performs three functions for the manufacturer in the bill of materials area. It maintains the bill of materials master file, prepares printed bills of materials, and explodes bills of materials for planning. TABS is coded in COBOL, making it easily adaptable to individual manufacturers' situations. It is available from the Honeywell Series 200 Users' Library by the regular request procedure.

Transportation

Honeywell has two application programs for the transportation industry. They are called ARS (Automatic Rating System) and *Schedule*.

ARS – performs the highly complex task of rating shipments for trucking companies. ARS handles the total rating job for commodity, class, and exception rates for the various tariff bureaus. Due to the variety of routes and types of shipments involved and the tight controls of the Interstate Commerce Commission and state regulating agencies, the task of rating manually is time-consuming and error-prone. ARS automatically applies the proper freight rate, as well as pertinent accessorial charges, to shipments. Rated bills, or PRO's can then be transmitted to the terminals responsible for delivering goods to consignees.

In addition to shipment rating, ARS enables transportation companies to improve operations by virtue of its accounting and management reports as well as by providing valuable input to other trucking and computer functions.

Schedule is a run-cutting program that performs the difficult task of manpower scheduling for local transportation runs. With *Schedule*, the task is simplified and expedited; the scheduler is given all possible alternative combinations of straight and split runs and pieces of work, each combination closely approximating a normal 8-hour work day. *Schedule* is extremely flexible, enabling it to adapt to the shifting schedules created by holidays, weather, and special events such as parades and sporting events.

Distribution

Honeywell's integrated systems for the distribution industry are based on three fundamental applications:

Sale – an order-processing application which enters orders, issues customer invoices and warehouse picking documents, and updates the appropriate accounting records.

CASH – Computerized Accounting System by Honeywell – an accounting application which integrates accounts receivable, accounts payable, and general ledger accounting, and also summarizes and analyzes data for management reports.

PROFIT – Programmed Reviewing, Ordering, and Forecasting Inventory Technique – a dynamic order-strategy application for inventory control and purchasing which optimizes inventory levels and replenishment orders at the most desirable customer service level.

Sale handles order processing, keeping track of goods at the warehouse by recording each item's identity, quantity, and bin location. This information is used to provide picking documents for warehousemen, customer invoices, and reorder information for the distributor's purchasing subsystem.

CASH is an integrated accounting system that concentrates on three areas: accounts receivable, accounts payable, and general ledger. From the transactions entered in these areas, CASH produces various sales and purchasing analyses as well as a number of weekly, monthly, quarterly, and annual reports.

PROFIT is an inventory management tool which enables the distributor to maintain, *at least cost*, the requisite inventory to support a desired level of customer service. Taking over the time-consuming and complex tasks involved in inventory management, PROFIT reviews the inventory records, determines when to order, and then issues the required purchase orders. PROFIT uses exponential smoothing, a statistical forecasting method, to estimate future demand based on an exponentially weighted moving average of past demands.

PROFIT determines the optimum inventory level for each item in the warehouse, using in its considerations the desired level of customer service, past demand history, previous levels of forecast error, vendor lead times, vendor discounts, and lot size. It also saves money by joint replenishment, by generation of economic order quantities, and by balancing purchasing and carrying costs to achieve the least total cost for the inventory.

The above applications may be used in building-block fashion to construct the core of a total system such as the one depicted in the illustration. These Honeywell-supplied applications can also be used separately or in various combinations as the user desires.

Banking

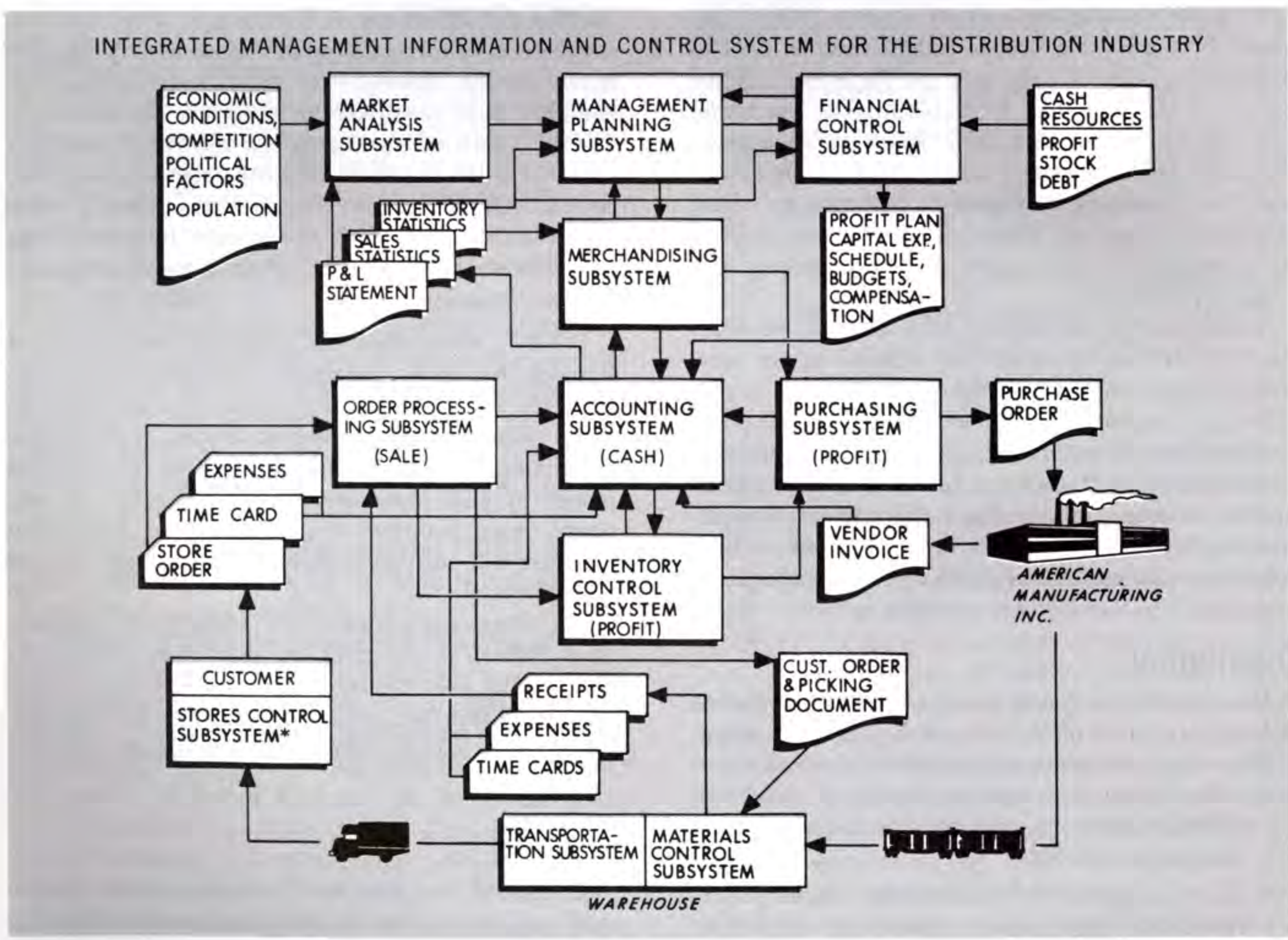
Honeywell has a series of comprehensive application packages for the banking community. Included are systems for automated proof and transit and for demand deposit, savings and mortgage loan, and instalment loan accounting.

A systematic approach to a management information system for the banking community is accomplished through Honeywell's multilevel bank processing concepts implemented in its series of comprehensive application packages serving this industry. The logical result of this approach is the generation of a central information file that provides banks with a completely integrated information system for timely and efficient operations. A central information file offers the flexibility for retrieval of customer information by using either a numeric or an alphabetic key. In addition, since the file contains all customer information, it is possible to obtain all customer bank interrelationships through inquiry into the file.

Automated Proof and Transit – Three programs are used to perform the automated proof and transit function.

The first program reads all magnetically encoded items received by the teller. As documents are read, all items are proven to their respective deposit tickets. During this process, a proof listing is generated to show any out-of-balance condition. In addition, all items are recorded on magnetic tape or disk. Up to ten pockets may be used for sorting, thus minimizing the number of sort passes required to process items.

The second program sorts all the items recorded on magnetic tape or disk by sending point to prepare for the creation of cash letters.



The third program generates appropriate cash letters and records all on-us items on a separate magnetic tape or disk for subsequent processing.

Demand Deposit Accounting – This system includes a MICR entry program as well as all other programs required to post accounts and produce required reports. The general design of this system makes it applicable to multibank operations.

The MICR entry program has been designed to minimize any problem in tailoring this program for any given situation. All input, output, print, and check-digit routines are in separate sections which can be replaced or modified.

The posting and reporting portion of this system provides for stop drafts, large checks, overdrafts, and all other exception situations encountered. Special services for large depositors are also provided. This system is suitable for small, medium, and large banks and lends itself to correspondent bank processing.

Savings and Mortgage Loan Accounting – Both on-line and batch processing are accommodated by this system. An on-line updating program is included which permits the posting of accounts through the use of Honeywell teller terminal equipment. The design of this system permits transactions from small branches (where the use of a Type 370 is not economically feasible) to be batch processed off-line.

The on-line updating program is modular in construction and consists of separate sections for input, processing, and output. The design of this system permits polling and nonpolling line techniques to be employed. Other terminal devices may be employed, if required.

Under control of the program, Series 200 computer systems accept, process, and transmit on-line messages as normal batch processing proceeds simultaneously. Through the use of the program interrupt feature of Series 200 computers, the program automatically interrupts batch processing, assumes control, and brings the teller message into computer memory. The program then retrieves the pertinent account record from the random-access file and processes transactions against the records while updating them. When this operation is completed, the program assembles the proper reply and initiates its transmission back to the inquiring terminal.

A series of posting and reporting programs is supplied which allow initial implementation of a batch processing system that can be expanded to an on-line posting system when desired, thus providing complete flexibility in implementing an on-line system.

Instalment Loan Accounting – This system is modular and may be altered to meet specific requirements of individual banks. However, in most cases the system will be found to be completely in line with the needs of any instalment loan accounting application.

System functions include: updating of present loan accounts; generation of new loan acceptance and rejection reports; formation of new loan contingent obligor files; printing of charge payment reports, rebate reports, paid-off account reports, doubtful/charge-off payments, final charge-off reports, due-day control reports, overdue notices, fines due reports, full transcripts, and daily reports of totals, and other reports. Along with the seemingly endless printing of internal and customer reports and statements, new loan coupon cards and new loan statements are produced. Every month, the computer analyzes the loan master file and uses the results to compile and print statistical reports.

Finance

Banks, brokers, investment fund trustees, mutual fund administrators, investment counselors, and insurance companies are the potential users of Honeywell's financial analysis application package. Anyone who invests in (or gives advice on) common stocks can benefit by taking advantage of the application package called *Computerized Portfolio Analysis.*

The programs in this package enable a subscriber to Standard & Poor's Compustat Tape Service to evaluate and compare the status and performance of several companies rapidly and simultaneously. The system is capable of calculating various important ratios and indicators of a company's growth, stability, and prospects. It also produces a company-versus-industry comparison, allowing an even more informed analysis to be performed. The analysis can also be used to select companies which meet criteria specified by the investment analyst. The programs perform the complete and lengthy calculations necessary to produce the analytical reports.

With Honeywell's portfolio analysis programs, investors and investment counselors can analyze and compare stocks efficiently and rapidly and thus gain a better basis for investment decisions.

Life Insurance

TIP (Total Information Processing) is a comprehensive set of 90 programs to serve the life insurance industry by handling policies from initial application to termination.

Subsystems include:

- New Issue
- Daily Cycle
- Periodic Production
- Annual Statement
- Reserves

New Issue is a set of programs for processing new applications, reinstatements, conversions, restorations, and exchanges. Among the many advantages to the customer is reduced "cost of acquisition." A major advantage is the ability to use the system for file conversion during TIP implementation.

The *Daily Cycle* performs daily production and maintenance of the policy master file, providing various reports and output files. Input processing includes new issues, premium remittances, and various externally generated transactions such as inquiries and file changes. Automatically generated processing includes billing, policy anniversary processing, changes in benefits and riders, and various other maintenance functions.

The *Periodic Production* subsystem includes accounting and commission runs, dividend updating, and policy exhibit.

The *Annual Statement* subsystem performs computations for generation of annual statement, along with valuation updating of the policy master file.

The *Reserve* subsystem generates minimum cash values, terminal reserves, and plan description rate tapes for all plans of insurance.

The TIP system operates on a low-cost, five-tape, 20K, Series 200 configuration with card reader, card punch, and printer. The broad selection of peripheral units in the Honeywell product line allows TIP to satisfy the data processing and cost requirements of a wide range of company sizes. In addition, the TIP user is also afforded transition to advanced systems under development.

Fire and Casualty Insurance

FACILE (Fire And Casualty Insurance Library Editions) is a total data processing system developed for the fire and casualty insurance industry. Among the more important data processing functions encompassed by FACILE are: handling of new business, policy changes, agent accounting, claims handling, etc. In addition, the system integrates several files into a single master file and prepares reports on an exception basis.

FACILE is open-ended and modular in design. It is an expandable system designed to provide optimal data processing with a variety of systems configurations.

Not only are a company's present business problems efficiently and effectively handled, but the basic building-block design of FACILE provides expansion to solve tomorrow's problems without compromising today's systems requirements. Through this philosophy, a company can select components applicable to today's particular operation and at the same time provide easy transitional steps to a larger and more powerful system as the company grows.

Education

Honeywell has developed a library of application systems designed to provide the nucleus of schools' total information requirements. Due to their broad areas of application, these systems produce the information necessary to evaluate current and plan future school operations, significantly improve pupil guidance practices, and make available a tool to be used in the instruction process itself. This is accomplished by focusing on pupil information, data banks for educational research, and instructional aids while also providing tools to enhance the school's administrative operations.

Pupil Information – In this area, Honeywell has developed an integrated set of subsystem designs for pupil assignment, attendance accounting, grade reporting, test analyses, and cumulative record maintenance. These application subsystems alleviate teachers' and administrators' paperwork burdens, automate pupil record maintenance and reporting, assist in the analysis of pupil data for use in career and curriculum planning, and develop a data bank for future evaluation and planning.

Registration data and permanent record information are processed in a cumulative record maintenance subsystem. By maintaining both active pupil record files and historical master files, this subsystem integrates all the pupil information applications. It ensures that any change in pupil data generated by any subsystem (test results, grades, attendance, etc.) is simultaneously available to all subsystems. In addition, in maintaining historical files, it creates an extensive data bank for educational research applications.

Class sectioning and pupil assignment, while significant manual jobs, represent a minor portion of a school system's data processing load. The ability to execute this application with one's own computer is of considerable significance. School systems may not only do a more efficient job of pupil assignment, but may also experiment with new concepts of schedule design by using these programs in simulation mode. Further, the results of the pupil assignment form the data base for other pupil accounting applications. Therefore, Honeywell has implemented two main series of pupil assignment programs to provide a wide range of capability. Each series is modular to permit the incorporation of unique user concepts.

An attendance-accounting subsystem relieves the teacher of clerical drudgery. While maintaining individual records for each pupil, this subsystem produces attendance registers and irregular-attendance reports, which automatically call significant patterns of absenteeism (along with relevant achievement data) to the attention of the appropriate guidance counselor. It also performs such accounting chores as calculating average daily and monthly attendance.

The grade-reporting subsystem processes grade data for inclusion in cumulative records and produces report cards, academic lists, and guidance reports—automatically or on request. This subsystem design also encourages the use of this data with that of the standardized achievement tests to produce student information for guidance counselors, highlighting any exceptional performance. It also produces special printouts such as labels and class lists.

Educational Research – Honeywell systems have an item of long-term significance in their facility for educational research. A permanent file, maintained on tape for each pupil from the day he first enters a school system, provides the bank of raw data necessary for a variety of data relationships, studies having important implications in the planning of pupil guidance, and future educational innovations on a system basis. Correlation studies and factor analyses relating items such as academic achievement and standardized test scores with indices of postgraduate academic and career success can be carried out effectively on Series 200 computers.

Instructional Aid – A most significant application of the computer in a school system is its use as an instructional device. EDP training is becoming vitally important in mathematics, science, business, and general education. Honeywell publishes a comprehensive set of programmed texts for training students in computer technology. In addition, a simplified programming language, WORDCOM,[1] has been implemented on the Series 200 for use in EDP survey courses.

Administrative Functions – Honeywell provides illustrative systems design to assist the educational user in automating his administrative operations, although recognizing that these systems must be adapted to satisfy each school system's operational environment.

A payroll system design, defining both manual and computer operations, which handles multiple classifications of payment and multiple deductions for insurance, professional society dues, credit unions, etc., is a major contribution to any school system's conversion to electronic data processing.

A system designed to facilitate the automation of library procedures is also a part of Honeywell's educational support. Implementation of this system reduces the clerical demands on professional librarians, freeing them to devote more time to assisting students, researchers, and other library users in identifying and locating sources of information.

Retailing

Honeywell's intensive engagement in the retailing industry covers all aspects of the field from capturing point-of-sale data to providing action-compelling management reports. An ever-growing set of application packages has been, and is being, developed under the closest cooperation with a number of nationally prominent retailers. Each package may be adapted by the retailer to suit his specific needs and each is a part of a flexible and expandable network. Following are brief descriptions of available retailing industry application systems:

Capture – Computer Automated Process Technique Using Register Entry – is the link that joins the point-of-sale acquisition of data with other building-block retailing subsystems. Honeywell's *Capture* edits, checks, and balances the source data provided by Uni-Tote[1] point-of-sale devices. It generates exception reports signaling error conditions and variances, produces register balance listings and control totals by type of transaction, and prepares validated magnetic tape input to accounts receivable, merchandise control, sales analysis, and statistical subsystems.

Approximately 80% of all the source data of a retailing total system originates at the point-of-sale. Viewed in this light, *Capture* is a keystone block in retailing total systems.

Accounts Receivable System – Programmed in COBOL, this system provides descriptive billing on a monthly cycle basis. The system accommodates 30-day, revolving, and fixed-payment accounts. It provides multilevel dunning, collection cards for accounts going into collection status as well as negative autho-

[1] Registered trademark of American Totalisator Company.

rization lists and automatic aging of accounts. A feature of the system is a special duplicate credit-issue program check.

Accounts Payable System – Also programmed in COBOL, this system provides for disciplined payments for merchandise and expense items by voucher check. It produces check registers, purchase journals, trial balances, aged listings, open prepay reports, open carrier claims, and expense journals. A vendor history file is maintained for vendor analysis.

Payroll – The payroll system consists of basic programs to produce pay checks, W-2 and 941A forms, personnel information reports, deduction registers, and quarterly history printouts. The system includes labor distribution reports and monthly reports of selling salaries. All programming is in COBOL to facilitate modification to suit individual requirements.

Applications currently under development include the following:

Staples Merchandise Control – This system provides automatic stock replenishment and forecasting within a 2-level central stock operation. It automatically reorders designated "action" items and generates sales/stock status reports on "report-only" items.

Fashion Merchandise Control – Sales analysis exception reports for fast sellers, slow sellers, and transfer candidates are produced by this system. It provides merchandise category/price range open-to-buy reports and generates class sales/stock status reports.

Big Ticket Inventory Control – This system provides on-line inquiry and processing of "big ticket" reservations and sales. The system features instantaneous data lookup and file updates through Visual Information Projection devices.

Data Processing Service Centers

For data processing service centers, Honeywell can provide a *Job Estimation and Control System* to help maximize profits through accurate job costing and control. Specific areas covered include:

- Systems analysis and design planning and control
- Machine time allocation and scheduling
- Systems utilization recording

GENPAY, a payroll system designed primarily for use by service centers, is sufficiently flexible to be accepted for use in an internal data processing system.

Honeywell also makes available *all* of its proven applications systems, spanning a wide range of business applications. These systems enable the data processing service center to tailor services to the exact requirements of its clients. Specific programs are described under industry headings in this chapter. General-purpose systems such as linear programming and network modeling packages, and others, are also available.

In addition to systems created by Honeywell, a Series 200 Users' Library provides a constantly growing source of programs which may be applied by data processing service centers.

Advantages of Honeywell applications systems coupled with the *Job Estimation and Control System* are underscored by the impressive percentage of independent EDP service centers which are already using Series 200.

[1] Honeywell's WORDCOM is an adaptation of the WORDCOM language found in Automatic Data Processing, second edition, by Gregory and Van Horn, published by the Wadsworth Publishing Company, Belmont, California.

8 Tables

Instruction Formats and Timing

Each Series 200 instruction is described in the following table in terms of its operation code, formats, and timing formulas. In addition, reference is made in each case to the page where the operations initiated by the instruction are described. The formulas given in the table provide execution times in memory cycles.

The internal operation of the Model 4200 processor differs from that of the other Series 200 processors in that data is moved in groups of four characters (a word) rather than singly. Consequently, the 4200 timing formulas differ considerably and are listed separately from those of the other processors.

Equivalent expressions for symbols used in the table are as follows:

A	Address of A-operand field.
B	Address of B-operand field.
D	One if there is a 2-bit overflow into LOR; otherwise zero.
N_1	Number of binary ones in a multiplier.
N_a	Number of characters in the field indicated by A.
N_{aw}	Number of words in the field indicated by A.
N_b	Number of characters in the field indicated by B.
N_{bw}	Number of words in the field indicated by B.
N_{b1}	Number of words that the A-field word occupies in the B field whether or not the operands characters have been modified by an arithmetic operation.
N_{b2}	Number of words in the B-operand field excluding N_{b1}

SYMBOL	MEANING
N_i	Number of characters in the instruction.
N_{ia}	Number of words in the items to be translated.
N_{ib}	Number of words in the result item.
N_{ic}	Number of information units (6-bit or 12-bit characters) to be translated.
N_j	Number of character locations bypassed to reach the next sequential op code.
N_m	Number of characters moved.
N_n	Number of bit positions shifted for automatic formatting.
N_r	Number of characters referenced.
N_{sc}	Number of characters scanned.
N_{sh}	Number of shifts.
N_{st}	Number of characters stored.
N_w	Number of characters in the A- or B-operand field, whichever is shorter.
N_{wj}	Number of words bypassed to reach the next sequential op code.
N_{ws}	Number of words stored.

SYMBOL	MEANING
n	Number of items in the table or the number of times the A operand is compared against some portion of the B operand.
V	Variant character.
W	Number of memory words used to store the data involved.
W_i	Number of 4-character words used to store one more than the total number of characters in the instruction.
X_0	Zero if no second scan (zero suppression); one if the scan is performed.
Y_0	Zero if no third scan (dollar-sign insertion); one if the scan is performed.
Z	Number of characters scanned during zero suppression.
Z_w	Number of words scanned during zero suppression.
$	Number of characters scanned during dollar-sign insertion.
$\$_w$	Number of words scanned during dollar-sign insertion.

Note: The timing formulas presented in the accompanying table are based on the use of direct addressing. If address modification is used, the formulas for the Models 120, 125, 200, 1200, 1250, and 2200 should be modified as follows:

1. *Indirect Addressing* – Add one memory cycle for each *character* extracted as a result of indirect addressing.
2. *Indexed Addressing* – Add three memory cycles for each indexed *address.*

Likewise, the use of address modification requires that the formulas for the Model 4200 be modified as follows:

1. *Indirect Addressing* – Add 1.16 memory cycles for each indirect *address* formed *plus* one memory cycle for each *word* extracted as a result of indirect addressing.
2. *Indexed Addressing* – Add 3.167 memory cycles if one address is indexed, 5.16 memory cycles if both addresses are indexed.

MNEMONIC OP CODE	INSTRUCTION NAME	INSTRUCTION FORMAT(S)	TIMING FORMULAS (Memory Cycles) 120/125/200/1200/1250/2200[1]		4200	DESCRIPTION PAGE
FIXED-POINT ARITHMETIC INSTRUCTIONS						
A	Decimal Add	A/A,B A/A A	No Recomplement[2] $N_i+2+N_w+2N_b$ Recomplement[2] $N_i+2+N_w+4N_b$		No Recomplement $W_i+N_{aw}+2N_{b1}+2.5N_{b2}+5$ Recomplement $W_i+N_{aw}+2.5N_{bw}+2N_{b1}+2.5N_{b2}+5$	31
S	Decimal Subtract	S/A,B S/A S	No Recomplement[2] $N_i+2+N_w+2N_b$ Recomplement[2] $N_i+2+N_w+4N_b$		No Recomplement $W_i+N_{aw}+2N_{b1}+2.5N_{b2}+5$ Recomplement $W_i+N_{aw}+2.5N_{bw}+2N_{b1}+2.5N_{b2}+5$	31
M	Decimal Multiply	M/A,B M/A M	See page 31 for representative times.		See page 31 for representative times.	31
D	Decimal Divide	D/A,B D/A D	See page 32 for representative times.		See page 32 for representative times.	32
BA	Binary Add	BA/A,B BA/A BA	$N_i+1+N_w+2N_b$		$W_i+N_{aw}+2N_{b1}+2.5N_{b2}+5$	32
BS	Binary Subtract	BS/A,B BS/A BS	$N_i+1+N_w+2N_b$		$W_i+N_{aw}+2N_{b1}+2.5N_{b2}+5$	32
ZA	Zero and Add	ZA/A,B ZA/A ZA	$N_i+1+N_w+N_b$		$W_i+2N_{b1}+N_{b2}+6$ S mode: Add 2 memory cycles	32
ZS	Zero and Subtract	ZS/A,B ZS/A ZS	$N_i+1+N_w+N_b$		$W_i+2N_{b1}+N_{b2}+6$ S mode: Add 2 memory cycles	32
SCIENTIFIC PROCESSING INSTRUCTIONS[3]			1200/1250	2200		
AMA	Floating Add	AMA/A,XY	$N_i+13+N_n/6$	$N_i+13+N_n/4$	$W_i+12+N_n/6$	32
AAA	Floating Add	AAA/XY	$10+N_n/6$	$10+N_n/4$	$W_i+7+N_n/6$	32
SMA	Floating Subtract	SMA/A,XY	$N_i+13+N_n/6$	$N_i+13+N_n/4$	$W_i+12+N_n/6$	32
SAA	Floating Subtract	SAA/XY	$10+N_n/6$	$10+N_n/4$	$W_i+7+N_n/6$	32
MAM	Floating Multiply	MAM/A,XY	$N_i+18+[N_1/6]+[N_n/6]$	$N_i+21+[N_1/4]+[N_n/4]$	$W_i+17+N_n/6$	33
MAA	Floating Multiply	MAA/XY	$15+[N_1/6]+[N_n/6]$	$19+[N_1/4]+[N_n/4]$	$W_i+12+N_n/6$	33
DMA	Floating Divide	DMA/A,XY	$N_i+25+N_n/6$	$N_i+31+N_n/4$	$W_i+20+N_n/6$	33
DAA	Floating Divide	DAA/XY	$21+N_n/6$	$29+N_n/4$	$W_i+15+N_n/6$	33
TAM	Store Floating Accumulator	TAM/A,X–	N_i+11	N_i+12	W_i+7	33
TAA	Store Floating Accumulator	TAA/XY	7	8	$W_i+4.5$	
TMA	Load Floating Accumulator	TMA/A,–Y	N_i+11	N_i+12	W_i+8	33
TAA	Load Floating Accumulator	TAA/XY	7	8	$W_i+4.5$	33
FBA	Floating Test and Branch on Accumulator Condition	FBA/A,XC	N_i+3 (No branch) N_i+4 (Branch)	N_i+3 (No branch) N_i+5 (Branch)	$W_i+4.3$ (No branch) W_i+5 (Branch)	33
FBI	Floating Test and Branch on Indicator	FBI/A,OD	N_i+2 (No branch) N_i+3 (Branch)	N_i+2 (No branch) N_i+4 (Branch)	$W_i+3.5$ (No branch) $W_i+4.2$ (Branch)	33
DTB	Decimal to Binary	DTB/A,–Y	N_i+20+D	N_i+24	W_i+17	33
BTD	Binary to Decimal	BTD/A,X–	N_i+21	N_i+24	W_i+12+W	33
TLM	Store Low-Order Result	TLM/A	N_i+10	N_i+11	W_i+8	34
TLA	Store Low-Order Result	TLA/–Y	6	7	W_i+4	34
TML	Load Low-Order Result	TML/A	N_i+10	N_i+10	W_i+9	34
TAL	Load Low-Order Result	TAL/X–	6	6	W_i+4	34
BMS	Binary Mantissa Shift	BMS/XM,V	$7+N_{sh}/6$	$8+N_{sh}/4$	$W_i+5+N_{sh}/6$	34
BIM	Binary Integer Multiply	BIM/A,B	$N_i+21+N_1/6$	$N_i+23+N_1/4$	$W_i+14.5$	34

MNEMONIC OP CODE	INSTRUCTION NAME	INSTRUCTION FORMAT(S)	TIMING FORMULAS (Memory Cycles) 120/125/200/1200/1250/2200[1]	4200	DESCRIPTION PAGE
			LOGIC INSTRUCTIONS		
EXT	Extract	EXT/A,B EXT/A EXT	N_i+1+3N_w	$W_i+3N_{b1}+7$	34
HA	Half Add	HA/A,B HA/A HA	N_i+1+3N_w	$W_i+3N_{b1}+7$	34
SST	Substitute	SST/A,B,V SST/A,B SST/A SST	N_i+4	$W_i+7.34$	34
C	Compare	C/A,B C/A C	$N_i+2+N_w+N_b$[4]	$W_i+2N_{b1}+N_{b2}+5.5$	35
B	Branch (Unconditional)	B/A	N_i+2[4]	$W_i+4.5$	35
BCT	Branch on Condition Test	BCT/A,V BCT	N_i+2[4]	$W_i+4.5$	35
BCC	Branch on Character Condition	BCC/A,B,V BCC/A,B BCC/A BCC	N_i+4	W_i+6	35
BCE	Branch if Character Equal	BCE/A,B,V BCE/A,B BCE/A BCE	N_i+4	W_i+6	35
BBE	Branch on Bit Equal	BBE/A,B,V BBE/A,B BBE/A BBE	N_i+4	W_i+6	35
			GENERAL CONTROL INSTRUCTIONS		
SW	Set Word Mark	SW/A,B SW/A SW	N_i+3[5]	W_i+5	36
SI	Set Item Mark	SI/A,B SI/A SI	N_i+3[5]	W_i+5	36
CW	Clear Word Mark	CW/A,B CW/A CW	N_i+3	W_i+5	36
CI	Clear Item Mark	CI/A,B CI/A CI	N_i+3	W_i+5	36
H	Halt	H H/A H/A,B H/A,B,V	N_i+2[4]	W_i+5	36
NOP	No Operation	NOP	N_i+2[6]	W_i+4	36
CAM	Change Addressing Mode	CAM/V CAM	N_i+2[6]	W_i+4	36
CSM	Change Sequencing Mode	CSM CSM/A CSM/A,B CSM/A,B,V	N_i+3[6]	W_i+4	36
SCR	Store Control Registers	SCR/A,V SCR/A SCR	N_i+5[4]	$W_i+W+4.33$ (Non-I/O Register) W_i+W+13 (I/O Register)	37
LCR	Load Control Registers	LCR/A,V LCR/A LCR	N_i+5[4]	$W_i+W+4.33$ (Non-I/O Register) W_i+W+24 (I/O Register)	37
LIB	Load Index/Barricade Register	LIB/A/B LIB/A	N_i+5[7] N_i+3[7]	W_i+2W+6 $W_i+W+4.5$	37
SIB	Store Index/Barricade Register	SIB/A/B SIB/A	N_i+5[7] N_i+3[7]	W_i+2W+5 W_i+W+4	37

MNEMONIC OP CODE	INSTRUCTION NAME	INSTRUCTION FORMAT(S)	TIMING FORMULAS (Memory Cycles) 120/125/200/1200/1250/2200[1]	4200	DESCRIPTION PAGE
			INTERRUPT CONTROL INSTRUCTIONS		
MC	Monitor Call	MC	N_i+2[6]	W_i+4	38
SVI	Store Variant and Indicators	SVI/V	$N_i+2+N_{st}+N_j$[8]	$W_i+N_{ws}+N_{wj}+8$	38
RVI	Restore Variant and Indicators	RVI/A,V	N_i+2+N_r[6]	W_i+7	38
RNM	Resume Normal Mode	RNM/A,B RNM/A RNM	N_i+3[9]	$W_i+4.5$	38
			DATA MOVE INSTRUCTIONS		
MCW	Move Characters to Word Mark	MCW/A,B MCW/A MCW	N_i+1+2N_w	$W_i+2N_{b1}+6$	38
LCA	Load Characters to A-Field Word Mark	LCA/A,B LCA/A LCA	N_i+1+2N_a	$W_i+2N_{b1}+5$	38
EXM	Extended Move	EXM/A,B,V EXM/A,B EXM/A EXM	N_i+1+2N_a	$W_i+2N_{b1}+5$	38
MAT	Move and Translate	MAT/A,B,V_1,V_2	N_i+3N_a[10]	$W_i+1.67N_{ia}+1.67N_{ib}+N_{ie}+6.5$	39
MIT	Move Item and Translate	MIT/A,B,V_1,V_2,V_3	$N_i+N_a+2N_{ie}$[10]	$W_i+1.67N_{ia}+1.67N_{ib}+N_{ie}+6.5$	39
MOS	Move or Scan	MOS/A,B,V MOS/A,B MOS/A MOS	$N_i+1+3(N_m)$[11] (Move) $N_i+1+3(N_{sc})$ (1200 Scan) $N_i+2+2(N_{sc})$ (2200 Scan)	$W_i+3+(3N_m+1)$	39
TLU	Table Lookup	TLU/A,B,V TLU/A,B TLU/A TLU	$N_i+1+n(N_a)+N_b$[11]	$W_i+n(N_{aw})+N_{bw}+3n+9$	39
			EDIT INSTRUCTION		
MCE	Move Characters and Edit	MCE/A,B MCE/A MCE	$N_i+1+N_a+2N_b+2Z+2\$$	$W_i+N_{aw}+2.3N_{bw}+2Z_w+2\$_w+6+X_o+Y_o$	39
			INPUT/OUTPUT INSTRUCTIONS		
PDT	Peripheral Data Transfer	PDT/A,C_1,C_2,...C_n	See description of PDT instruction.	See description of PDT instruction.	40
PCB	Peripheral Control and Branch	PCB/A,C_1,C_2,...C_n	See description of PCB instruction.	See description of PCB instruction.	40

(1) Add one memory cycle to these formulas when calculating Model 2200 times, except where the formula is followed by footnote (4), (9), (10) or (11).

(2) Subtract one memory cycle from this formula if the instruction is being executed in a Model 120, 125, 1200, or 1250 processor.

(3) These formulas apply only to Model 1200, 1250, and 2200 processors; the scientific unit is not available with the Models 120, 125, and 200.

(4) Add two memory cycles to this formula if the instruction is being executed in a Model 2200 processor.

(5) Subtract one memory cycle from this formula if the instruction is being executed in a Model 1200 or 1250 processor in the format Op Code/A,B.

(6) Subtract one memory cycle from this formula if the instruction is being executed in a Model 1200 or 1250 processor.

(7) This formula applies only to Model 1200, 1250 and 2200 processors; this instruction is not available with the Models 120, 125, and 200.

(8) Subtract one memory cycle from this formula if the instruction is being executed in a Model 200 processor.

(9) Add two memory cycles to this formula if the instruction is being executed in a Model 2200 processor; subtract one cycle from the formula if the instruction is being executed in a Model 1200 or 1250 processor.

(10) If the instruction is being executed in a Model 2200 processor, do not add the one memory cycle mentioned in footnote (1).

(11) This formula applies only to Model 1200, 1250, and 2200 processors; this instruction is not available with the Models 120, 125, and 200.

Correspondence Among Series 200 Central Processor, Card, and Printer Codes

Key Punch	Card Code	Central Processor Code	Octal	High Speed Printer	Key Punch	Card Code	Central Processor Code	Octal	High Speed Printer
0	0	000000	00	0	$\overline{0}$ or —	X, 0 or X(1)	100000	40	—
1	1	000001	01	1	J	X, 1	100001	41	J
2	2	000010	02	2	K	X, 2	100010	42	K
3	3	000011	03	3	L	X, 3	100011	43	L
4	4	000100	04	4	M	X, 4	100100	44	M
5	5	000101	05	5	N	X, 5	100101	45	N
6	6	000110	06	6	O	X, 6	100110	46	O
7	7	000111	07	7	P	X, 7	100111	47	P
8	8	001000	10	8	Q	X, 8	101000	50	Q
9	9	001001	11	9	R	X, 9	101001	51	R
	8, 2	001010	12	,		X, 8, 2	101010	52	#
#	8, 3	001011	13	=	$	X, 8, 3	101011	53	$
@	8, 4	001100	14	:	*	X, 8, 4	101100	54	*
Space	Blank	001101	15	Blank		X, 8, 5	101101	55	"
	8, 6	001110	16	>(2)		X, 8, 6	101110	56	≠(2)
	8, 7	001111	17	&	— or $\overline{0}$	X or X, 0(1)	101111	57	½ or !(2)(3)
&/0 or &	R, 0 or R(1)	010000	20	+		8, 5	110000	60	<(2)
A	R, 1	010001	21	A	/	0, 1	110001	61	/
B	R, 2	010010	22	B	S	0, 2	110010	62	S
C	R, 3	010011	23	C	T	0, 3	110011	63	T
D	R, 4	010100	24	D	U	0, 4	110100	64	U
E	R, 5	010101	25	E	V	0, 5	110101	65	V
F	R, 6	010110	26	F	W	0, 6	110110	66	W
G	R, 7	010111	27	G	X	0, 7	110111	67	X
H	R, 8	011000	30	H	Y	0, 8	111000	70	Y
I	R, 9	011001	31	I	Z	0, 9	111001	71	Z
	R, 8, 2	011010	32	;		0, 8, 2	111010	72	@
.	R, 8, 3	011011	33	.	,	0, 8, 3	111011	73	,
□	R, 8, 4	011100	34	)	%	0, 8, 4	111100	74	(
	R, 8, 5	011101	35	%		0, 8, 5	111101	75	CR
	R, 8, 6	011110	36	■		0, 8, 6	111110	76	□(2)
& or &/0	R, or R, 0(1)	011111	37	?(2)		0, 8, 7	111111	77	¢(2)

(1) Special Code (for use with H-400/1400 and H-800/1800 cards). The second (alternative) card code is equivalent to the stated central processor code when control character 26 is coded in a card read or punch PCB instruction.
(2) Indicates symbol which will be printed by a printer which has a 63-character drum (Type 222 printers).
(3) The exclamation point replaces the one-half symbol on a type roll containing the Mark II character font.

HONEYWELL
ELECTRONIC
DATA
PROCESSING
WELLESLEY HILLS,
MASSACHUSETTS 02181

102776801