GA22-7011-5
File No. S370-01

Systems

IBM System/370 Model 158 Functional Characteristics

IBM

Sixth Edition (September 1978)
This is a major revision of, and obsoletes, GA22-7011-2, -3, -4 and the following Technical Newsletters:

GN22-0483 (dated April 2, 1974)
GN22-0490 (dated March 3, 1975)
GN22-0504 (dated December 17, 1975)
GN22-0518 (dated October 7, 1976)
GN22-0522 (dated November 4, 1976)

Changes have been made throughout this manual. A technical change to the text or to an illustration is indicated by a vertical line to the left of the change.

Changes are periodically made to the information herein; before using this publication in connection with the operation of IBM equipment, refer to the *IBM System/370 Bibliography,* GC20-0001, for editions that are applicable and current.

Publications are not stocked at the address given below; requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for reader's comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Product Publications, Department B98, PO Box 390, Poughkeepsie, NY, U.S.A. 12602. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Preface

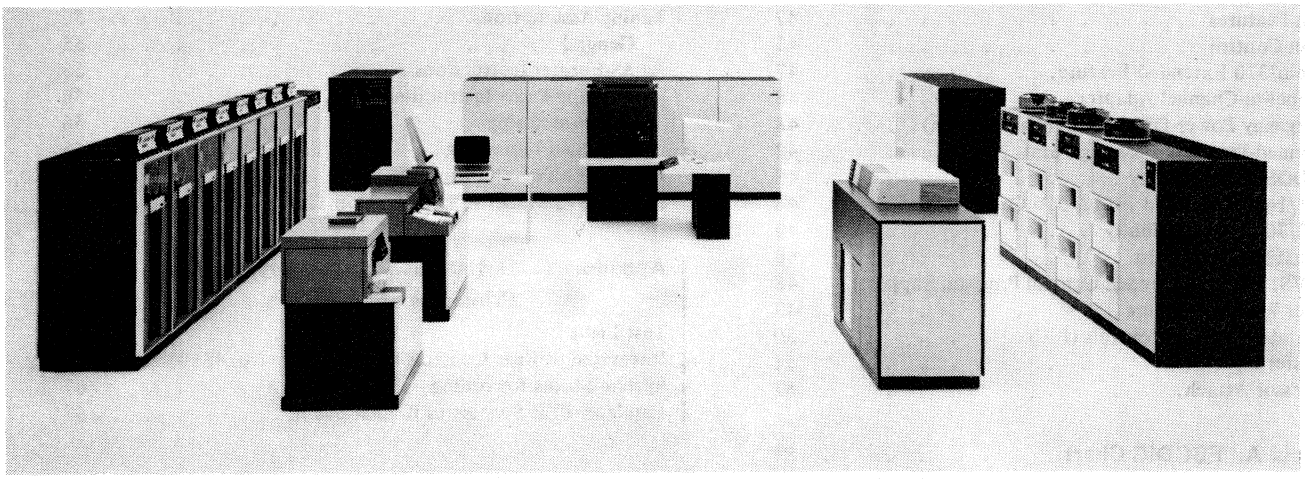
This manual describes the functional characteristics and features of the IBM System/370 Model 158. It provides management, programming, and operations personnel experienced in System/370 operation with a fundamental understanding of the Model 158. *IBM System/370 Principles of Operation*, GA22-7000, is referenced throughout the text and should be used in conjunction with this publication.

Only information that is of particular concern to the Model 158 user is discussed in this manual. Discussion covers the central processing unit and storage functions, the system console, the console display, and the IBM 3213 Console Printer, as well as a summary of standard facilities and optional features.

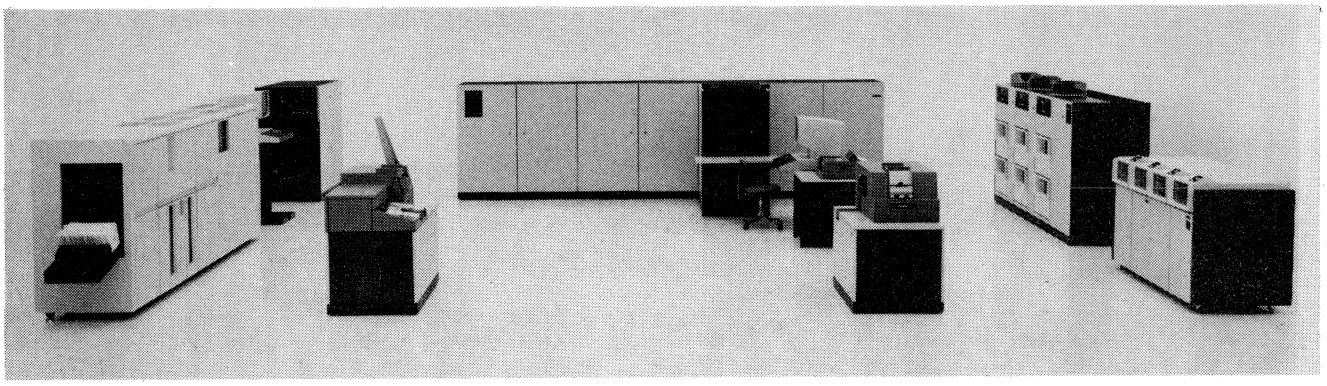
In addition, the "Hierarchical Monitoring System," "Deviations from *System/370 Principles of Operation,"* and the System/370 extended feature are discussed.

Detailed information on channel characteristics for System/370 Model 158 is available in *IBM System/370 Model 158 Channel Characteristics*, GA22-7012.

Contents



IBM System/370 Model 158 (Representative Uniprocessor Configuration) (Design Model)



IBM System/370 Model 158 (Representative Attached Processor Configuration) (Design Model)

The IBM System/370 Model 158 is a medium-size, high-performance data processing system that extends and enhances the basic System/370 concept. The Model 158 incorporates many advanced features such as integrated monolithic main storage, virtual storage capability, enhanced timing facilities, integrated storage controls, and a display console. These functional capabilities encourage expanded applications for all large-scale users.

The basic components of the System/370 Model 158 are the CPU (the IBM 3158 or 3158-3 Processing Unit), the display console, and an optional console printer. Included within the 3158 and 3158-3 are the processor, main storage, buffer storage, at least one byte-multiplexer channel, and two block-multiplexer channels. Input/output (I/O) devices are attached to the channels via control units.

The 3158 and 3158-3 are available in eight main storage capacities:

3158	*3158-3*	*Main Storage Capacity*
I, AP1, MP1	U31, A31, M31	524,288 bytes (512K)
J, AP2, MP2	U32, A32, M32	1,048,576 bytes (1,024K)
JI, AP3, MP3	U33, A33, M33	1,572,864 bytes (1,536K)
K, AP4, MP4	U34, A34, M34	2,097,152 bytes (2,048K)
KJ, AP5, MP5	U35, A35, M35	3,145,728 bytes (3,072K)
L, AP6, MP6	U36, A36, M36	4,194,304 bytes (4,096K)
LJ, AP7	U37, A37, M37	5,242,880 bytes (5,120K)
LK, AP8	U38, A38, M38	6,291,456 bytes (6,144K)

A high-speed buffer is standard; its storage capacity is 8,192 bytes (8K) in the 3158, and 16,384 bytes (16K) in the 3158-3.

The 3158-3 is an advanced version of the 3158, which can be converted to a 3158-3. Information within this manual applies to both models unless noted otherwise. Differences between the two are discussed where relevant.

For input/output operations, one byte-multiplexer channel (channel 0) and two block-multiplexer channels (channels 1 and 2) are standard. Optional features provide for up to three additional block-multiplexer channels (channels 3, 4, and 5). Alternatively, with channel 3 installed, channel 4 may be used as a second byte-multiplexer channel. For details, see Figure 1.

VIRTUAL STORAGE

Regardless of the real storage capacity, when the Model 158 is operating in extended control (EC) mode with dynamic address translation invoked, all logical addresses within the 24-bit addressing scheme of System/370 are available. Therefore, the maximum logical (virtual) address is 16,777,215. System/370 Model 158 is designed primarily to support this virtual storage environment; it is not subject to restraints normally imposed on programming applications by the amount of available storage, and the operational flexibility of the installation is enhanced.

MULTIPROCESSOR SYSTEM

Multiprocessing is a logical extension of multiprogramming. A Model 158 multiprocessor (MP) system uses two multiprocessing versions of the 3158 or 3158-3, joined by an IBM 3058 Multisystem Unit. The CPUs can execute programs simultaneously while sharing total system resources, including data. Benefits derived from multiprocessing include operational efficiency and flexibility and improved system availability. The most critical component of the MP system is the control program. CPUs are considered to be system resources as are I/O devices and storage.

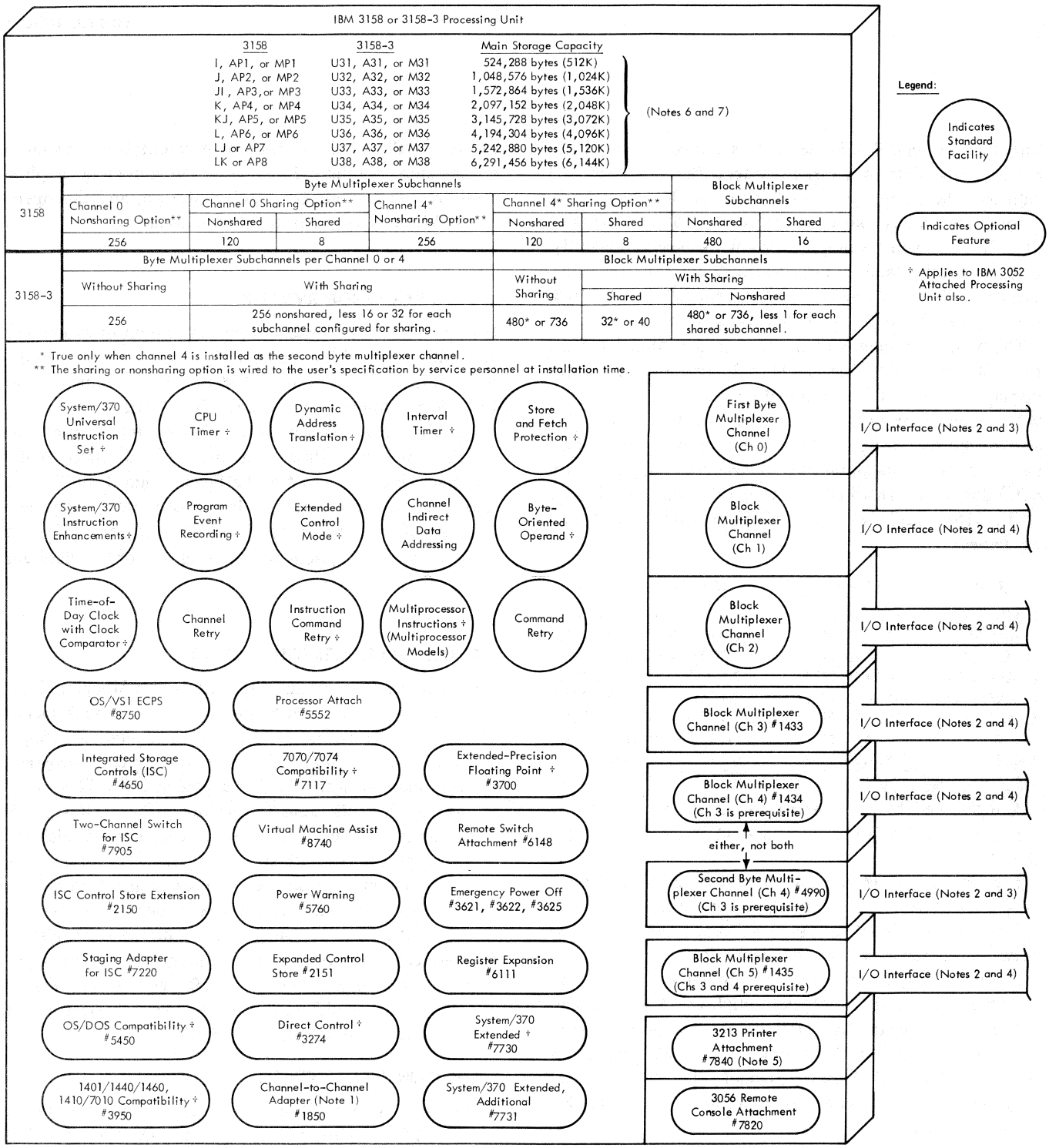
ATTACHED PROCESSOR SYSTEM

The instruction-processing capabilities of a Model 158 can be significantly increased by adding an IBM 3052 Attached Processing Unit (APU) Model 1 to an upgraded 3158 or 3158-3 Processing Unit (CPU). The APU executes programs concurrently with the CPU and accesses the CPU's main storage. All I/O functions are performed by the host 3158. This tightly coupled configuration, called the Model 158 attached processor (AP) system, is designed for the user who requires instruction execution capability beyond that available with the uniprocessor system but whose needs do not presently justify a Model 158 multiprocessor system. The attached processor system can provide internal performance that is generally 50 to 80% greater than that of a uniprocessor Model 158 with identical job streams and configurations running under MVS 3.7.

PROGRAMMING SUPPORT

Extended control (EC) mode with dynamic address translation (DAT) is supported by OS/VS1, OS/VS2, VM/370, and DOS/VS. Multiprocessing is supported by OS/VS2 (MVS). Attached processing is supported by the AP first-customer-shipment release level of OS/VS2 (MVS) modified by a selectable unit. Extended storage (5,120K and 6,144K) is supported by OS/VS2 for both uniprocessor and attached processor systems and by VM/370 and OS/MVT for uniprocessor systems only.

In addition, OS/360 and DOS/360 (in hardstop mode) provide support in System/370 basic control (BC) mode.



Notes:

1. The channel-to-channel adapter (one per system, attached to byte or block multiplexer channel) permits interconnection of two channels. One control unit position on the Model 158 channel can connect to one control unit position on any other System/370 or System/360 channel. Only one adapter is needed per connection; it counts as one control unit on both channels.
2. Attaches as many as eight control units; available subchannels are shown on chart.
3. Operates in either burst mode or byte mode; multiplexing capability on bytes, groups of bytes, or blocks.
4. Operates in burst mode only; multiplexing capability on blocks or multiple blocks. Available subchannels are shown on chart.
5. The 3213-1 Printer is optional in display mode, mandatory in printer-keyboard mode.
6. A multiprocessor system consists of two multiprocessing models of 3158 or 3158-3 connected by a 3058 Multisystem Unit. The storage capacities of both CPUs must be equal and no more than 4,096K in any system using two 3158s; they need not be equal in a system using two 3158-3's if the main storage of each CPU is other than 512K or 1,536K. Maximum main storage capacity is 8,192K.
7. An attached processor system consists of an IBM 3052 Attached Processing Unit (with an IBM 3056 Remote System Console) connected to a Model A-series 3158. The 3052 contains no main storage or channels.

Figure 1. Model 158 Configurator

COMPATIBILITY

Any program written for System/370 will operate on the Model 158 provided that it:

1. Does not depend on system facilities (storage size, I/O equipment, optional features, and so forth) not included in the configuration. For example, because the Model 158 has volatile main storage, the program must not depend on the validity of data in main storage after system power has been turned off and then restored.
2. Is not time dependent.
3. Does not depend on results or functions defined in *IBM System/370 Principles of Operation,* GA22-7000, to be unpredictable or model dependent.
4. Does not depend on results or functions that are defined in this publication to be deviations from those described in the *IBM System/370 Principles of Operation,* GA22-7000. See Appendix C for a description of the deviations.
5. Does not depend on the absence of system facilities that are included in the Model 158. For example, it should not depend on interruptions caused by the use of operation codes or command codes that are not assigned or are not installed.
6. Does take into account, if it uses the read direct and write direct instructions that these instructions use real instead of logical addresses when the invalidate page table entry instruction is installed.

REMOTE SUPPORT FACILITY

If the system malfunctions, this facility provides the customer with the combined expertise of local and remote IBM System/370 Model 158 support specialists. This is achieved by linking the system, via a teleprocessing network, to the field technical support center. The advantages of this support technique are evident; however, the customer's data security and privacy are subject to some exposure while the remote support facility is in use.

The security measures enlisted to ensure minimal exposure are as follows:

1. The customer's security (console) key is required in order to display the teleprocessing link frame.
2. The teleprocessing link is established only after thorough verification of the identity of both the customer and the remote support center.
3. Each mode of remote support (remote program, remote monitor, and remote control dedicated) is initiated and identified at the customer's installation. When the teleprocessing link is established for remote monitor or remote control dedicated mode, a header message containing the maintenance strategy level for the customer is displayed onsite and at the remote support center. These levels, which are determined by the customer and altered only at his request, are:
   a. Customer allows concurrent maintenance.
   b. Customer allows stand-alone maintenance only.
   c. Security account: the system must be cleared of all customer data prior to remote support, including making devices that contain security data not-ready.
4. Every operation initiated at the remote center can be monitored by the customer.
5. The teleprocessing link can be disconnected at any time at the customer's installation by pressing the remote pushbutton on the control panel.

One of three modes of operation can be selected at the discretion of the service representative and with the customer's approval.

1. *Remote Program*: This mode is identical to the RETAIN/370 link and provides all of the online test executive program (OLTEP) security facilities. For example, to protect against accidental modification of customer data, OLT(S)EP diagnostic programs restrict writing to noncustomer volumes or to designated areas of customer volumes. Also, to protect against disclosure, OLT(S)EP diagnostic programs read/transmit the smallest amount of data that permits satisfactory diagnosis.
2. *Remote Monitor*: This mode allows the remote center to monitor the data being displayed on the console display. The monitoring may take place concurrently with the customer's operation, which therefore displays customer data to the remote center.
3. *Remote Control Dedicated*: In this mode, all console functions, both service and operational, are available to the remote center. The customer console is inoperative; however, it is possible for the customer to terminate the operation by pressing the remote pushbutton. It should be noted that all data sets on the system could be accessed by the remote center; however, every operation initiated at the remote center can be customer-monitored. This mode will normally be used with the system dedicated to the service personnel, although it is possible to use it concurrently with customer operation.

All modes of operation are independent of operating system release levels. This facility is not dependent on the CPU, I/O, or channels being operational.

STANDARD FACILITIES AND OPTIONAL FEATURES

For a description of the standard facilities and optional features for the Model 158, see "Facilities and Features Summary."

The central processing unit (CPU) contains the elements required to decode and execute the instructions and emulator programs featured on the system. Most CPU functions are under microprogram control.

Machine cycle time for the CPU is 115 nanoseconds.

Instruction Fetch (I-Fetch)

The CPU contains instruction-fetch (I-fetch) components and controls that allow prefetching of instruction data from main storage. For the I-fetch function, instruction buffers enable most I-fetches to overlap the execution time of previous instructions.

I-fetch sequences may overlap CPU operations if main storage is not busy when the fetch sequence starts. CPU and I/O operations overlap after a channel is started and until a break-in occurs. Because I/O operations share CPU data paths to and from storage, CPU operations are suspended when an I/O storage request occurs.

Instruction Retry

The ability to recover from intermittent failures and thereby increase the reliability of the Model 158 is provided through retry techniques. Microprogram routines save source data before it is altered during an operation, thus making instruction retry possible. When an error is detected, a microprogram routine returns the CPU to the beginning of the operation or to a point in the operation that was correctly executed; the operation is then resumed. Retry procedures use both additional system logic and the retry microprograms.

Most operations in the basic system are retriable. A machine-check error during I-fetch causes the I-fetch to be retried. The manner in which the instruction is retried depends on the instruction. Some instructions do not change the original data in the registers until the last cycle of execution; these instructions are retried from the beginning. Other instructions change source data in the registers and are retried from a checkpoint, using the intermediate results.

Instruction retry operates on all but four instructions: diagnose, test and set, read direct, and write direct. If an error occurs during the execution of an I/O instruction, the execution is checked to determine whether the retry threshold has been passed. If the instruction execution has not passed this threshold, the instruction is retried automatically, without program assistance. A machine check interruption is taken at the completion of a successful retry for recording purposes.

If the instruction execution has progressed too far to be retried, an I/O interruption is taken, or the condition code is set to indicate that a CSW and limited channel logout (LCL) have been stored because the I/O operation was not started. The appropriate device-dependent error recovery routine can be scheduled to take the required recovery action. Usually, if an error in the execution of the start I/O instruction occurs before the I/O device becomes involved on the I/O interface, instruction retry is still possible.

Error Checking and Correction

Every data path in the CPU is parity-checked by byte, either directly or indirectly. The adder is parity-checked in three levels: halfsum, carry, and fullsum checks. Every data path between the CPU and main storage is also parity-checked. Error correction codes apply to data stored in, and fetched from, main storage; single- and double-bit error detection and single-bit error correction are performed.

Timing Facilities

Timing facilities include the time-of-day (TOD) clock, the interval timer, the CPU timer, and the clock comparator, all of which are defined in *IBM System/370 Principles of Operation*, GA22-7000.

Machine-Check Handling

Machine-check handling is described in *IBM System/370 Principles of Operation,* GA22-7000.

The starting location of the Model 158 machine-check extended logout (MCEL) area, which is variable, is recorded in the three low-order bytes of control register 15. The starting location is set to 512 (decimal) after a system reset operation, but it can be changed by the user. The length of the MCEL area for the Model 158 can be found by using the store CPU ID (STIDP) instruction, which stores the length value in an accessible area of main storage. The length value does not exceed 672 bytes.

The I/O extended logout pointer is not used on the Model 158. The store channel ID (STIDC) instruction executed on the Model 158 will always indicate the length of the longest I/O extended logout area as 0 and the channel model number as all zeros (hexadecimal).

Resets

The detailed effects of each of the following reset functions are shown in Figure 2:

- Power On
- System Clear
- Initial Program Reset (IPR)
- Program Reset (PR)
- Hardware Reset
- Check Reset
- PSW Restart

Area Affected	Reset Function						
	Power On	System Clear	Initial Program Reset (Load)	Program Reset	Hardware Reset	Check Reset	PSW Restart
Control Storage	L	U	U	U	U	U	U
TOD Clock	C	U	U	U	U	U	U
Main Storage	C	C	U	U	U	U	U
Keys in Storage	C	C	U	U	U	U	U
General Registers	V	V	V	V	U	U	V
Floating-point Registers	V	V	V	V	U	U	V
Control Registers	I	I	I	V	U	U	I
PSW	C	C	C	V	U	U	P
CPU Timer	C	C	C	V	U	U	C
Clock Comparator	C	C	C	V	U	U	C
CPU State	S	S	S	S	S1	S2	S
Attached Channels	R	R	R	R	M	U	R
Buffer Storage	C	C	C	C	U	U	C
TLB and Index Array	X	X	X	X	U	U	X
Prefix Register	C	C	C	V	U	U	C

Legend:

C	Contents are set to zero with valid parity.
I	Contents are set to the initial state.
L	The initial microprogram load function is performed.
M	Content of channel local stores and bump is unchanged; all channels set with channel check and the interface is reset.
P	PSW at location 0 is used.
R	The channel is reset and signals a system reset on the I/O interface.
S	Execution of the current CPU operation, if any, is terminated; pending interruptions and machine check conditions are cleared. The CPU enters the stopped state.
S1	Execution of any current CPU operation is terminated.
S2	Pending errors are cleared.
U	Contents including parity are unchanged.
V	Parity for the contents is made valid.
X	Valid bits are reset; OK bits are set.

Figure 2. Detailed Effects of Reset Function

Reloadable Control Storage (RCS)

The 8,192 words (72 bits each) of reloadable control storage contain microprograms to control the basic processor and channel functions and features, such as emulators. RCS is implemented in monolithic circuit logic and is loaded from the console (load) file, by the user, as an initial microprogram load (IMPL) procedure; it is not available for programming purposes. The CPU and the channels share RCS. The CPU and each channel operate within their own microprograms, and they share CPU logic by switching control at specified points in the microprograms. This change of control is called "break in." When a break in occurs, the current microprogram is temporarily halted while another microprogram is given control. The address of the interrupted microprogram is retained until control is returned to it.

Local Storage

There are two local storages; one is used exclusively by the channels, and one is shared by the CPU and channels. CPU local storage contains the control, general, and floating-point registers, as well as certain unit control word (UCW) storage and control areas. I/O local storage is used for data buffering on the block multiplexer channels and for working areas on the byte multiplexer channels.

Main Storage

The Model 158 uses monolithic main storage, which is addressed and controlled by the storage control unit (SCU). Main storage sizes vary from 512K to 4,096K. Contained within the first 512K of main storage is a 16K array extension used for unit control word (UCW) storage. Monolithic storage is volatile; data is not preserved when power is interrupted.

The main storage cycle time (from the standpoint of the CPU) varies with the operation according to the following:

Operation	*Cycle Time*
Read, 8 bytes	1,035 ns
Read from buffer, 4 bytes	230 ns
Read from buffer, 8 bytes	345 ns
Fast write, 8 bytes	690 ns
Partial write	920 ns

Buffer Storage and Index Array

Buffer storage contains four 4K compartments. Each compartment is divided into 16-byte halfblocks. The buffer storage access width is that of a doubleword (eight bytes). Space in the buffer is reserved on a block basis (32 bytes), and is loaded one halfblock at a time. A buffer halfblock is assigned when it is fetched from main storage and set into the buffer.

The buffer assignment algorithm uses six algorithm bits located in the index array. There are six bits for every block address (four entries); these bits are interrogated for all CPU and translation fetches. The status of these bits and of the halfblock valid bits determines the precise location of the buffer assignment.

The following are 3158 characteristics that are different from the 3158-3:

- Buffer storage contains two 4K compartments, rather than four.
- Space is reserved on a halfblock (16-byte) basis, rather than on a block basis.
- The buffer assignment algorithm uses a single least recently used (LRU) bit rather than six algorithm bits.
- There is one LRU bit for every block and halfblock address (two entries), rather than six bits for every block address (four entries).

The index array maintains the addresses of the contents of the upper and lower buffer compartments. The index array is interrogated during each CPU storage reference to determine if the referenced data is in the buffer. The index array is cycled at the referenced block address (both block and halfblock addresses on the 3158), and the four entries (two on the 3158) read out are compared to the referenced row and valid bit. An equal comparison of the entries (row and validity) determines that the buffer compartment contains the wanted data. An initial program reset or initial program load (IPL) sets the index array row addresses to valid parity, turns off all valid bits, and turns on the OK bits.

The system can continue to operate even when a component in the buffer fails. When a failure is detected, the OK bit for that block is turned off, and all subsequent fetches for that block are made directly to main storage. A machine check occurs when a block in the buffer is deleted for error-recording purposes.

Virtual Storage

Virtual storage is an image of 16,384K (16,777,216 bytes) created by the control program. It is functionally equivalent to the real or main storage of the system, and its contents are physically located in main storage or auxiliary disk storage.

Storage Allocation

Real (main) storage is allocated by the system control program on a dynamic paging basis. An entire program need not be resident in main storage throughout its execution. Instead, page size (2K or 4K) portions are transferred into main storage (in any available location) only as needed. Each page of main storage becomes available for reallocation to another job or task upon completion of the task, when the task is no longer using that page, or upon request for that storage space by a higher priority task.

Translation Lookaside Buffer (TLB)

The translation lookaside buffer (TLB) is to address translation as the buffer is to storage. Needless retranslations are avoided because the TLB stores up to 128 logical-to-real pairs of address translations. Each entry in the TLB represents either a 2K or a 4K area, depending on the page size specified.

CPU storage references requiring translation are first compared with the TLB to determine whether a current translation exists. If one exists, the corresponding real address in the TLB is used, as described in "Buffer Storage and Index Array." If no translation exists, translation is performed and the result is stored in the TLB.

Translation generates real addresses for the segment and page table entries. The fetching of these entries interacts with the buffer storage and index array like any other CPU storage reference. A new translation replaces a previous translation at that entry.

The TLB is purged of all entries:

1. When a purge TLB (PTLB) instruction is executed.
2. When a load control (LCTL) instruction modifying a segment or page size in control register 0 (CR0) is executed.
3. When the enter key on the keyboard is pressed.
4. When the store control character is touched with the light pen.
5. When a machine check occurs.
6. When a program reset is executed.
7. When an initial program reset is executed.

A partial purge of the TLB occurs if an LCTL instruction modifies CR1.

Channel Operations

The channels share control storage and data flow paths with CPU functions to perform I/O operations. Many channel functions are performed by the shared hardware, using a break-in technique to service I/O requests for the CPU and main storage.

The channels store and fetch all data to and from main storage, using real addresses. Channel data does not enter the buffer (see Figure 3). When a channel stores data in main storage, the index array is interrogated; if the data corresponding to that storage address is in the buffer, the valid bit for the buffer entry is turned off.

Since the channels do not implement dynamic address translation, CCWs in virtual storage must be translated by the control program before they are executed. To allow the designation of contiguous areas of virtual storage to be mapped into noncontiguous areas of real storage, indirect data addressing is provided. For complete channel information, see *IBM System/370 Model 158 Channel Characteristics*, GA22-7012, and *IBM System/370 Principles of Operation*, GA22-7000.

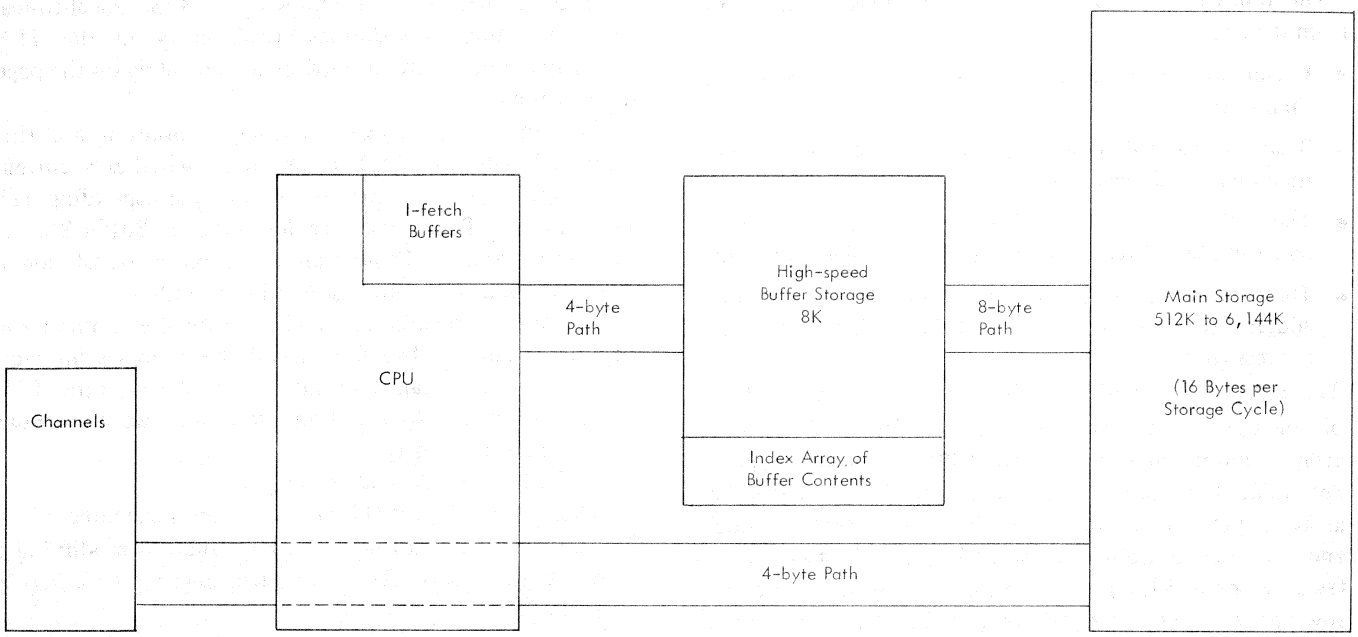


Figure 3. Conceptual Data Flow in Model 158

Program Event Recording

Program event recording (PER) is a valuable programming debugging aid and provides for the recording of information about selected program events as the events occur. PER is enabled by turning on bit 1 of the EC mode program status word (PSW). Control registers 9-11 control the selection of registers and storage locations. For details concerning PER, see *IBM System/370 Principles of Operation,* GA22-7000.

Store Status Facility

The store status facility allows control information to be preserved and stored after a reset operation. The facility may be selected by touching the light pen to the Store Status control character on the manual frame immediately following the reset function.

The system console of the Model 158 provides facilities for operating and controlling the system and for displaying system status. A small control panel, a keyboard, a cathode-ray tube (CRT) display, and a console disk file constitute the major operational components of the console. A console printer is available.

Most of the switches and pushbuttons normally associated with an operator's console are replaced by appropriate designations that are selectively displayed on the console screen. The entire system status cannot be displayed at one time; information is grouped logically into various frames (display categories), which are displayed on the CRT one at a time. Most of these frames are used as service aids by service personnel; only four (configuration, manual, program and alter/display) are used by the operator. Thus, advanced system maintenance and diagnostic procedures are combined with ease of operation to provide a high degree of efficiency.

The system console functions in one of two modes, depending on the operating system console support mode selected at initial microprogram load (IMPL) time. In systems providing device independent display operator console support (DIDOCS), the console can operate in display mode as a graphics display console with keyboard and selector (light) pen input and CRT output. The optional 3213 Printer provides hard-copy output. Alternatively, printer-keyboard mode may be selected with or without DIDOCS support, and the console then functions as a console I/O device similar to a 3215 Printer-Keyboard. When the console is operating in printer-keyboard mode, system-operator messages appear on the display screen but they can be displaced by subsequent messages before being acknowledged by the operator. The primary means of system communication in printer-keyboard mode are the 3213 Printer and the console keyboard.

The console file, located adjacent to the kneespace under the console keyboard, provides the microprograms required for the initial microprogram load (IMPL) of the reloadable control storage (RCS) of both the CPU and the console processor. A similar file, located in the rear of the console, is used for service and diagnostic purposes. Switches for the channel-to-channel adapter (CTCA) and for the integrated storage controls (ISC) are also located in the console file enclosure. An integrated data adapter for remote support of the Model 158 is standard on the console (see "Remote Support Facility" in the Introduction).

REMOTE SYSTEM CONSOLE

Another console, the IBM 3056 Remote System Console, is available as a special feature. It allows users to have operator control of a Model 158 from as much as 150 feet from the CPU. This enables users to have CPUs away from the center of activity, and thus permits more flexible organization of space for increased efficiency and convenience. For example, the 3056 can be located closer to the other system components which require attention. This console can be particularly worthwhile to users with multiple System/370s, enabling them to centralize operations.

The 3056, mounted on a standalone base, contains a CRT display and keyboard, identical to those on the main console. However, it has no light pen or printer. This console provides the operator with remote operational control over the system, including the ability to IPL, display the maintenance frames, and use the alter/display function that enables the contents of the following segments to be altered or displayed:

- Real Storage
- Virtual Storage
- Storage Keys
- Real Channel UCW
- Logical Channel UCW
- Active UCW's
- CPU Local Storage
- I/O UCW Local Storage
- I/O Buffer Local Storage
- Control Registers
- General Registers
- Foating-point Registers
- PSW

Controls not provided at the remote system console include those for system power on, IMPL, TOD clock, lamp test, and system power off.

The 3056 does not connect to a channel, but instead is a slave to the main console, and operates in parallel with it. The CRTs show the same displays, and responses can be made from either keyboard.

The remote system console has a disable key switch similar to that on the main console. When the main console key is locked and the remote system console key is unlocked, the remote operator has unrestricted access to all frames. When the remote system console key is locked, the remote operator has access only to the program frame, and is prohibited from performing an IPL or from altering programs or data. When the main console key is unlocked, the remote system console is inoperative.

CONSOLE SUPPORT

Device independent display operator console support (DIDOCS) for the CRT is the same as that provided for the IBM 3277 Display Station Model 2 supported by OS Release 21. This includes the selector pen (also referred to in this manual as the light pen), the keyboard, and the display area of twenty-four 80-character lines (the twenty-fifth line is used by system functions and is independent of DIDOCS).

The 3213 Console Printer is supported as an output-only console under multiple console support (MCS) or its equivalent.

Console support is provided under DIDOCS and MCS only when the display console is operating in display mode.

DISPLAY MODE CONSIDERATIONS

Basic to an understanding of console operation in display mode is an understanding of the cursor, the console buffer, and the definition of display data.

Cursor

The cursor is a positional indicator that assists the operator in entering data into the system accurately. Cursor positioning is controlled by keyboard operation, by light-pen operation of cursor controls (on the manual and alter/display frames), or by program control.

Console Buffer

The console buffer can store 2000 characters. Each buffer location corresponds to a similar screen location. (For example, buffer location 0 corresponds to display position 0.) Alphameric characters, certain special and control characters, and cursor information can be stored in any buffer location. If character codes other than those defined in bold outline in Table 1, Appendix A, are transferred to the buffer, the data characters displayed are not defined.

Data Fields

Console buffer storage, display, and logical manipulation of data are all field-oriented. A field is a group of consecutive alphameric characters starting with an attribute character and terminating with another attribute character which, in turn, defines the beginning of the next field. An attribute character is an eight-bit character that defines the characteristics of the data field that follows it. Programmed entry of attribute characters establishes a formatted display. Figure 4 shows the characteristics indicated by the setting of the bits of the attribute character.

Provision is made for defining data as protected or unprotected; for tagging data as having been modified; for controlling the brightness of displayed fields regulating the use of the selector pen, and inhibiting display of data.

The attribute character is discussed under "Programmed Operations" (see "Display Mode").

CONTROL PANEL

The control panel of the System/370 Model 158 is shown in Figure 5. The CPU and service usage meters are located immediately beneath this panel.

Emergency Pull Switch

Pulling this switch turns off all power to the system proper (except for emergency power-off controls), and all control units and I/O devices connected to the I/O power-control interface. When pulled, the switch latches in the out position and disables the power-on pushbutton. The switch can be restored to its normal position by servicing personnel only. When the emergency pull switch is in the out position, the power-on pushbutton is not effective.

Attribute Character Bits	X*	1	Unprotected/ Protected	Not Used	Intensity– Selector Pen Detection		Not Used	Modified Data Transfer
Bit Position	0	1	2	3	4	5	6	7

*Determined by the setting of bits 2, 4, 5, and 7.

Figure 4. Attribute Character Format

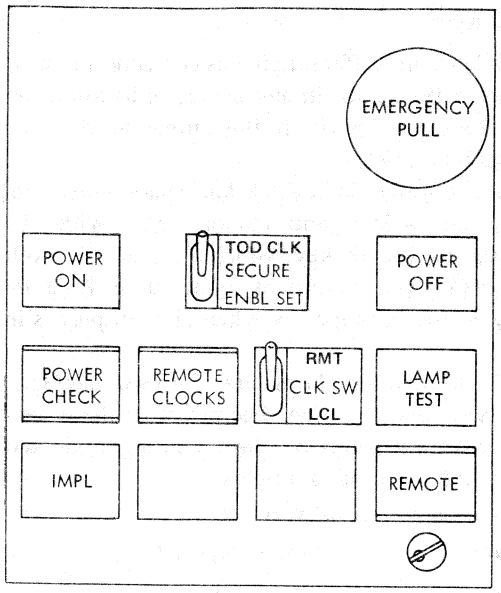


Figure 5. Control Panel, System/370 Model 158

Power-on Pushbutton

The power-on pushbutton is pressed to initiate the system power-on sequence. As part of this sequence, a power-on reset is performed. No instructions or I/O operations are executed until the system is explicitly directed to do so. After successful completion of the power-on sequence, the system enters the stopped state, the storage-protection array is cleared, and the TOD clock is cleared with its status forced to the not-set state.

The power-on pushbutton is backlighted red during the power-on sequence. If a power check condition exists, the pushbutton continues to be backlighted red. On successful completion of the power-on sequence, this pushbutton is backlighted white. Following a successful power-on sequence, an initial microprogram load (IMPL) is automatically initiated.

During the power-on sequence, some noncritical circuits may be in an unstable condition and may be temporarily activated. This does not affect the power-on sequence or the subsequent power-on reset.

Power-off Pushbutton

The power-off pushbutton is pressed to initiate the system power-off sequence. The contents of main storage are not preserved.

TOD Clock Switch

This spring-return lever switch provides an interlock against the inadvertent setting of the time-of-day clock. The set clock instruction is executed only while the switch is in the ENABLE SET position.

IMPL Pushbutton

Pressing this pushbutton causes the initial microprogram loading of the system console and CPU reloadable control storage (RCS). During IMPL, the message 'IMPL IN PROCESS' is displayed on the CRT. At completion of the load, a system reset is automatically executed.

Lamp Test Pushbutton

Pressing the lamp test pushbutton lights the power check and remote indicators, and the light emitting diodes (LED) on the console display bezel.

Remote/Local Clock Switch and Remote Clock Indicator

This switch and its associated indicator are used in a 3158-3 multiprocessing system. Their operation and function are discussed under "System Control Panel Features."

Remote Pushbutton/Indicator

The remote pushbutton/indicator lights when the remote adapter is online. Pressing this pushbutton when the indicator light is on halts the teleprocessing function instantly (see "Remote Support Facility" in the Introduction).

Power Check Indicator

The power check light comes on to indicate that either a circuit breaker or thermal switch is tripped. During the power-on sequence, the power check light comes on to indicate that a voltage has failed to reach the required level. *All cases require the attention of service personnel.*

KEYBOARD

With the system console in display mode, the keyboard can duplicate most of the light pen (selector pen) functions. The exceptions are the selection of a program function key (PFK) entry (see "Program Frame") and resetting the flashing ALARM indication, both of which require light-pen selection. With the system console in printer-keyboard mode and the program frame displayed, the keyboard is the only means the operator can use to communicate with the system.

The three types of keys on the keyboard are: alphameric, cursor control (including backspace, forward and backward tabulate), and function.

Alphameric Keys

In conjunction with the shift keys, the 45 alphameric keys are capable of generating 63 EBCDIC codes. All keys have momentary action, with the exception of the keys with a 'T' in the top right corner (see Figure 6). These typamatic keys can be held down for repetitive use.

Typed characters appear on the CRT at the cursor location unless the console is operating in display mode and the cursor is in a protected field or in an attribute-character location. Keyed-in data is not effective until ENTER is pressed. This greatly reduces the possibility of entering erroneous or incorrectly formatted data.

Cursor Control Keys

Cursor control keys are differentiated as character-oriented keys that position the cursor in any character location and field-oriented keys that position the cursor at the first character location of a field.

The five character-oriented keys are backspace, cursor up, cursor down, cursor left, and cursor right. With the exception of the backspace key, which functions in both display mode and printer-keyboard mode, these keys are operative only in display mode or when alter/display is in use.

The three field-oriented keys are forward tabulate, backward tabulate, and new line. The effect of these keys on cursor movement depends on the operating mode and on the program definition of data fields.

Cursor Control Keys

Character Oriented		*Field Oriented*	
←	Backspace	→|	Forward Tabulate
↑	Cursor Up	|←	Backward Tabulate
↓	Cursor Down	↵	New Line
←	Cursor Left		
→	Cursor Right		

Backspace: This key causes the cursor to move backward (to the left) one character position each time the key is pressed.

Cursor Up [↑]*:* Pressing this key causes the cursor to move up the screen in the same column. From the top of the screen, the cursor wraps to the bottom of the same column and continues upward again.

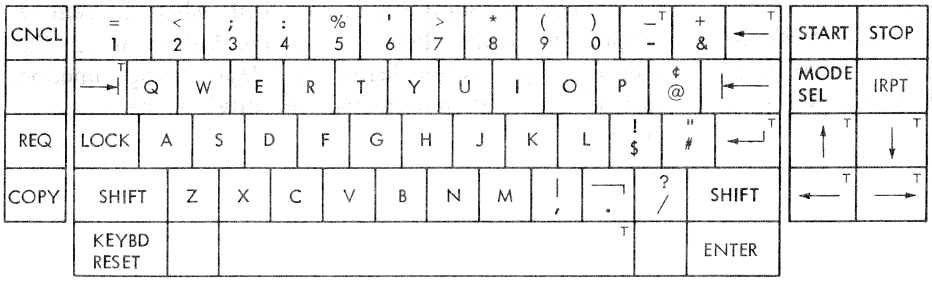


Figure 6. Console Keyboard

Cursor Down [↓] : Pressing this key causes the cursor to move down the screen in the same column. From the bottom of the screen, the cursor wraps to the top of the screen in the same column and continues downward again.

Cursor Left [←] : Pressing this key causes the cursor to scan to the left, progressing up the screen one line at a time. From the top left corner of the screen, the cursor wraps to the bottom right corner of the screen and continues scanning.

Cursor Right [→] : Pressing this key causes the cursor to scan to the right, progressing down the screen one line at a time. From the bottom right corner of the screen, the cursor wraps to the top left corner of the screen and continues scanning.

Forward Tabulate [⇥] : With the console in display mode, the cursor spaces to the first character location of the next unprotected field. If the buffer is unformatted or if there are no unprotected fields, the cursor is repositioned to character location 0. When the console is in printer-keyboard mode, or when the alter/display frame is in use, pressing this key generates a no-operation (NOP).

Backward Tabulate [⇤] : Three sets of circumstances are recognized.

1. With the console in display mode, the cursor is located in either the attribute character or the first alphameric character of an unprotected field or in any character of a protected field.
   Action: The cursor moves to the first alphameric character of the first preceding unprotected field.
2. With the console in display mode, the cursor is located in any character of an unprotected field other than the first.
   Action: The cursor moves to the first character location of the field.
3. The buffer either contains no protected fields or is unformatted.
   Action: The cursor is positioned at character location 0.

When the console is in printer-keyboard mode, or when the alter/display frame is in use, pressing this key results in a no-operation.

New Line [↵]: With the console in display mode, cursor action is as follows:

1. (Formatted Buffer)–The cursor moves to the first unprotected character position in the first line down to have such an unprotected position. If there are no unprotected fields, the cursor moves to character location 0.
2. (Unformatted Buffer)–The cursor moves to the first position of the next line.

When the alter/display frame is in use, the current display is rolled up one line.

When the console is in printer-keyboard mode, the cursor moves to the first position of the next line.

Function Keys

Function keys are differentiated as those with mode-independent functions (i.e., those keys that function in the same manner each time they are used, regardless of whether the console is operating in display or printer-keyboard mode), and those with mode-dependent functions. In the first category are the start, stop, mode-select, interrupt, copy, lock, and shift keys; in the second category are the cancel, request, keyboard-reset, and enter keys. Mode-independent functions are described in the following text. Mode-dependent functions are illustrated in Figure 7.

Start: With the system in the manual (stopped) state, pressing the start key initiates CPU processing. The key is ineffective, however, if the system is in a hard-stop error condition.

Stop: Pressing the stop key places the system in the manual state.

Mode Select: If the mode select key is pressed while the program, alter/display, or configuration frame is displayed, the manual frame appears. If the mode select key is pressed while the manual frame is displayed, the program frame appears.

Interrupt: Pressing the interrupt (IRPT) key presents an external interruption to the CPU.

Copy: Pressing the copy key causes the displayed frame (with the exception of the program frame) to be copied on the console printer, if available.

Lock: Pressing the lock key locks the shift keys down. The shift lock is released when either of the two shift keys is pressed.

Shift: Pressing either of the two shift keys causes the upper character of those keys having two characters, or the uppercase of single-character keys, to be effective.

CONSOLE DISPLAY

The console display is a cathode-ray tube unit with a minimum number of operator controls. The operator controls consist of manual intensity controls, a security key, and various bezel indicator lights (see Figure 8).

Key	Display Mode	Printer-Keyboard Mode	Alter and Display Mode
CNCL (Cancel)	Presents an attention interruption to the channel. The attention identifier (AID) byte is set to CANCEL.	Presents a unit exception interruption to the channel if a read command is in process. Data in the buffer is transferred to main storage.	Resets the alter/display frame to its initial state.
REQ (Request)	Generates a no-operation (NOP).	Presetns an attention interruption to the channel.	Generates a no-operation (NOP).
KEYBD RESET (Keyboard Reset)	Resets an inhibit keyboard condition, attention identifier (AID) characters, and a pending attention interruption if the keyboard is not disabled because of an I/O operation in progress.	Generates a no-operation (NOP).	Causes an erase input function (nullifies all alterations displayed on the screen.).
ENTER	Presents an attention interruption to the channel. The attention identifier (AID) byte is set to ENTER.	Data keyed into the console buffer is transferred to main storage.	Causes data keyed into the console buffer to be transferred to main storage, and terminates function and facility specifications selected on the alter/display frame.

Figure 7. Mode-dependent Function Keys

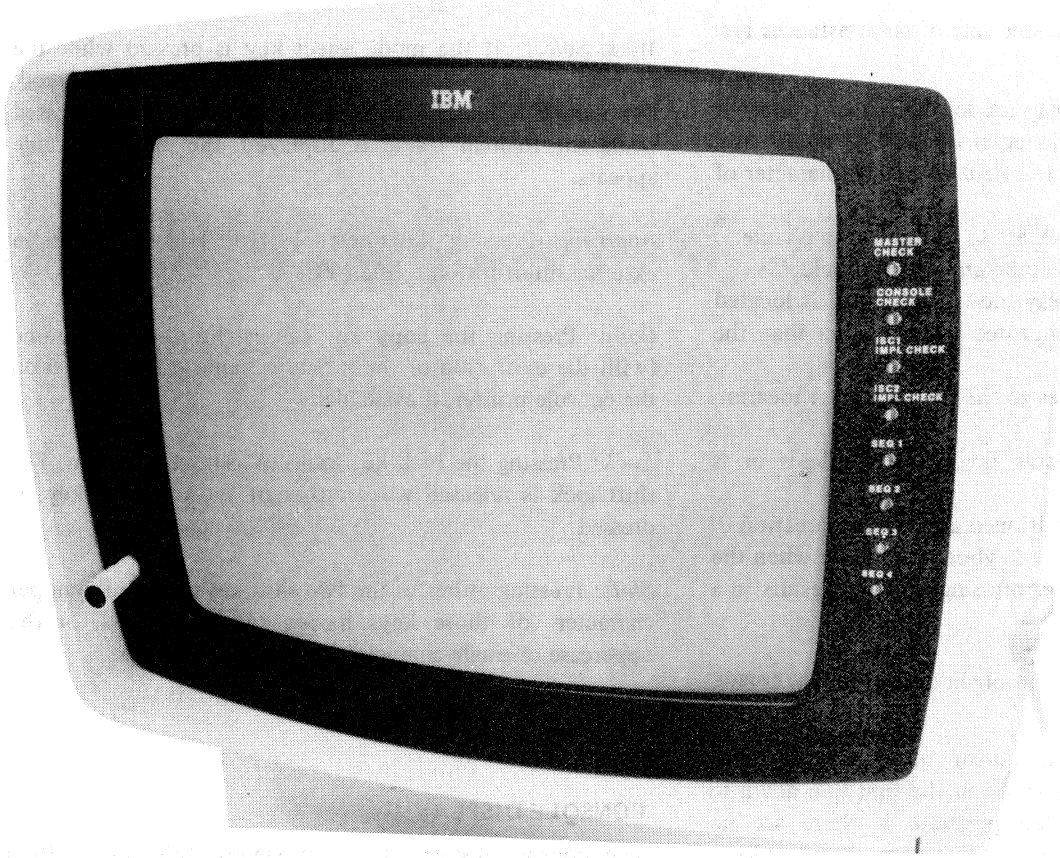


Figure 8. Console Display

Intensity Controls

The intensity controls are on the left side of the bezel. The outer control varies the intensity of all displayed characters, both normal intensity and high intensity characters. The inner control varies the intensity of characters displayed in normal intensity fields; it has no effect on character fields programmed for high intensity display.

Security Key

The security key lock is on the right side of the console display. With the security key removed or in the vertical position, system data security is placed under control of system programming, and the operator is prevented from leaving the program frame. With the key positioned horizontally, the mode select, frame select, and interrupt keys are no longer disabled; and the operator is free to display other frames.

Bezel Indicators

- Master Check–Turned on by any error in the system except integrated storage controls (ISC) IMPL checks.
- Console Check–Turned on by any error in the console.

If a console error occurs, console IMPL is automatically reinitiated and the message 'IMPL IN PROCESS' is displayed. The frame displayed at the completion of the IMPL depends on which frame was displayed at the time of the error. If the program frame was displayed, the frame reappears without any messages that may have been present prior to the console error. If any other frame was displayed, the service frame (with the console error indications) is displayed.

- ISC1 IMPL CHECK–ISC1 did not complete IMPL.
- ISC2 IMPL CHECK–ISC2 did not complete IMPL.
- Sequences 3 and 4 lights on (1 and 2 off)–Console file diskette not mounted.
- Sequence 2 light on (1, 3, and 4 off)–Console security switch not enabled.

All other sequence lights or combinations of sequence lights have no operational significance; they are used for diagnostic purposes.

Display Frames

Operation of the Model 158 requires an understanding of four frames on which the system configuration, operator manual controls, the alter/display function, and messages and system status are displayed. On the configuration, manual, and alter/display frames, either light pen (selector pen) or keyboard selection may be used. For example, the rate switch appears on the manual frame as:

▪ R–RATE
  ▪ 1–PROCESS
  ▪ 2–I-STEP

Either process rate or instruction-step rate can be light-pen-selected simply by pressing the tip of the pen against the control character ▪ (lozenge) preceding the option. Keyboard selection is made by typing the letter R followed by 1 or 2, depending on the option required. An arrow indicates the current selection.

Cursor controls are provided on frames where positioning of data being entered into the console buffer is required. These controls are discussed in the description of each frame when appropriate.

Configuration Frame

At the completion of initial microprogram load (IMPL), which is normally initiated by the power-on sequence, the configuration frame automatically appears (see Figure 9). Installed features, channel configuration, and a list of shared unit control words (UCW) are among the items displayed on this frame. If printer-keyboard mode of operation is required, it must be selected, since the power-on sequence takes place in display mode. Timer options are also selected on the configuration frame.

Exit from this frame is made either by light pen selection of the 'manual' control character (bottom left corner) or by pressing the mode select key on the keyboard. The manual frame then appears.

Selecting 'service' (bottom center) will display the service frame, which is not required for normal operation. Selecting 'copy' results in a facsimile of the configuration frame being printed on the console printer, if the console printer is available.

Manual Frame

The manual frame (Figure 10) displays the current state of the Model 158 and provides a means of initiating operator functions, using either the light pen (selector pen) or the keyboard.

Other frames are selected from the manual frame. In normal operation, the operator may request either the program frame so that output messages can be received, or the alter/display frame to examine the registers. Selection of the alter/display frame, for instance, is made by pressing the light pen against the control character preceding the selection or by entering the characters 'F3' from the keyboard.

Operator functions that can be selected by light pen or keyboard action are:

O1–PSW Restart (Prgm)
O2–Restart (Prgm)
O3–System Reset
O4–Load
O5–Store Status
O6–Sys Reset (Clear)
O7–Load (Clear)

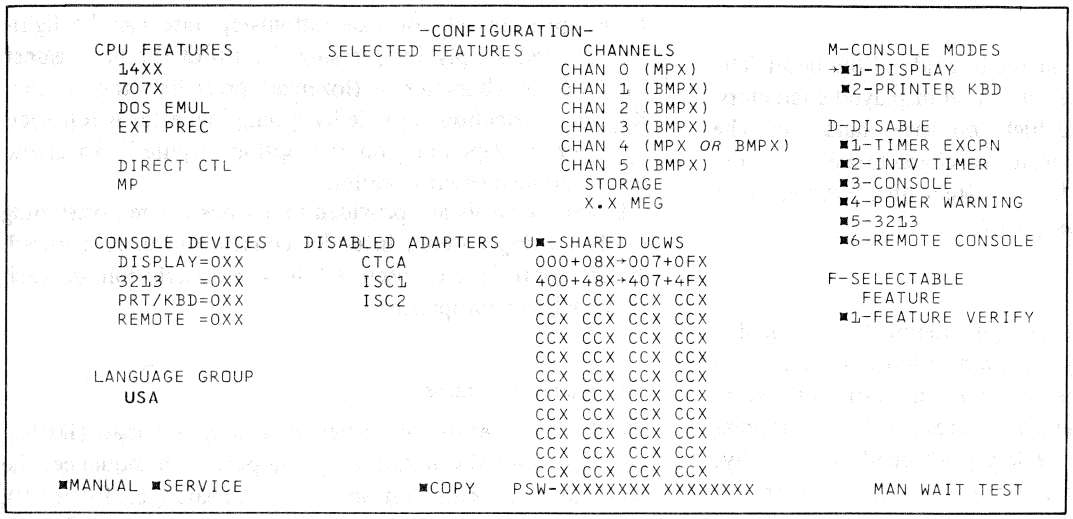


Figure 9. Configuration Frame (3158-3 shown; 3158-1 differs slightly)

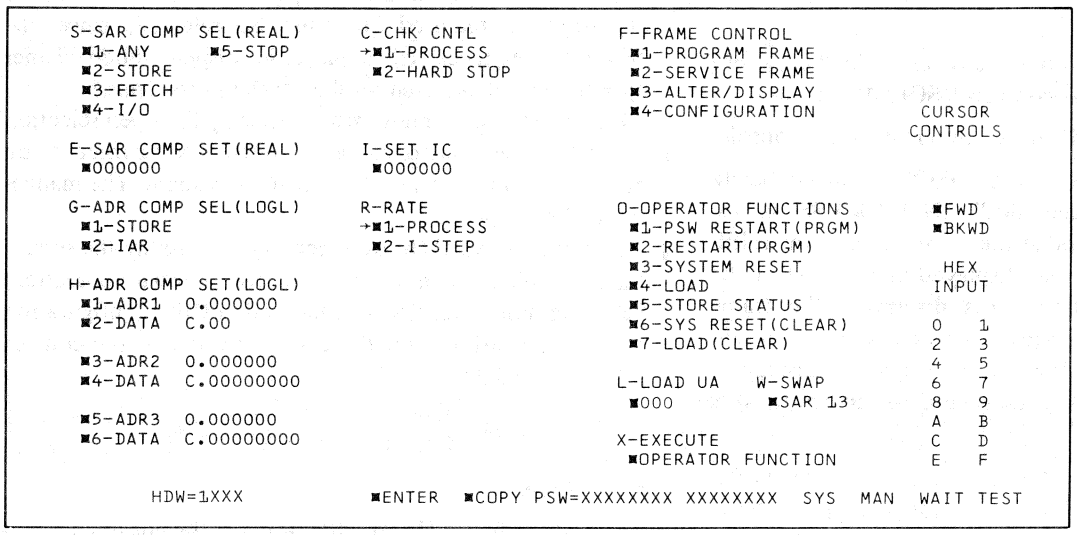


Figure 10. Manual Frame

After the operator selects the desired function, having previously entered the load unit address for load or load (clear), the execute (X) operator function must be invoked, either by typing 'X' or by pressing the light pen to the control character.

O1–PSW Restart (Prgm): Selecting PSW RESTART (PRGM) causes a system reset followed by loading of a new PSW from location 0. If instruction-step rate is in use, the manual loop is entered prior to executing the first instruction. If process rate is in use, processing resumes with the new PSW, and the program frame is displayed.

O2–Restart (Prgm): When RESTART (PRGM) is activated, the current instruction is completed if the system is running. The updated PSW is stored in location 8. A new PSW is loaded from location 0, and processing continues, using the new PSW. If the system is stopped but not hardstopped, it enters the running state. The program frame is automatically displayed.

O3–System Reset: Initiation of SYSTEM RESET causes the CPU clocks to stop in order to permit all pending storage operations to be completed. Then all machine logic is reset. A branch to the reset microprogram is forced, all channel operations are terminated, and a reset is signaled to all control units. All registers, buffers, the index array, and local storage become validated.

O4–Load: When LOAD is selected, followed by L–LOAD UA, a cursor appears under the first digit position of load unit address. The light pen can then be used to select a hexadecimal digit from the hexadecimal input matrix, or the keyboard can be used. The selected digit appears in the

first position of the load unit address, and the cursor advances to the next position. This process continues until the initial program load (IPL) device address has been entered. Upon successful completion of IPL, the program frame appears automatically. If the IPL is unsuccessful, the manual frame remains.

O5–Store Status: The selection of STORE STATUS immediately following the reset function allows control information to be preserved and stored.

O6–System Reset (Clear): SYS RESET (CLEAR) initiates the same action as RESET but, in addition, main storage and storage protection arrays are cleared to zeros with good parity.

O7–Load (Clear): LOAD (CLEAR) initiates the same action as LOAD but, in addition, the IPL is preceded by an initial program reset and the clearing of main storage and storage protection arrays.

W–Swap SAR 13: SWAP SAR 13, when selected, causes bit 13 of the storage address register (SAR) to be inverted. This results in changing the physical location of storage addresses, and the function can be used to load programs even though storage errors may exist. This function, when selected, is not activated until a load (clear) or system reset (clear) is performed.

The SAR compare select and address compare select functions are used to stop the system when certain predetermined conditions are met.

Bits 29-31 of the SAR are ignored for CPU and I/O data transfers of less than four words. Bit 28 is also ignored in printer-keyboard mode on four-word data transfers. The logic does not distinguish between main storage and storage protection array transfers.

S–SAR Compare Select (Real): Selection of one of these functions causes the address in the storage address register (SAR) to be compared with the address in the compare register, the address of which is set by E–SAR COMP SET (REAL). If a match occurs and the selected condition is met, the CPU will either stop or send out a sync pulse, depending on the setting of the stop function, S5. The SAR compare select functions are described in the following text.

S1–ANY – Causes the action when SAR matches the compare register (real addresses).

S2–STORE – Causes the action when the SAR matches the compare register and a store operation is being performed (real addresses).

S3–FETCH – Causes the action when the SAR matches the compare register and a fetch operation is being performed (real addresses).

S4–I/O – Causes the action if the SAR matches the compare register (real addresses) and the console is operating in printer-keyboard mode.

E–SAR Compare Set (Real): The SAR compare parameter is changed by selecting the control character to the left of the address field. The new address appears left-justified as it is entered from either the hex input matrix or the keyboard. The operation is completed by selecting the enter control character or the enter key on the keyboard.

G–Address Compare Select (Logical): When either G1–STORE or G2–IAR (instruction address register) is selected, the stopping hardware is activated by the ADR COMP SEL and ADR COMP SET parameters (G and H). The cursor indicates the selected facility. (Note: Selection of G1 or G2 degrades system performance.)

H–Address Compare Set (Logical): The address compare parameter is changed by selecting the control character to the left of the desired field or by typing the required letter and number combination (e.g., H1). The cursor then appears to the right of the decimal on the selected line. Up to three address bytes may now be entered. Backspacing the cursor over the decimal allows a base register value to be entered. When 'enter' is activated, the address is right-justified.

Selection of the H2 parameter positions the cursor to the right of the decimal on the data line. The state of the data parameter may be set by backspacing the cursor over the decimal and entering one of the following:

C (default)	Stop condition* on an equal compare.
1	Stop condition* if selected bits = 1.
0	Stop condition* if selected bits = 0.

*Stop condition means that, if the corresponding condition is true, a stop *may* occur, subject to the remaining stop conditions being true.

When an IAR or STORE compare occurs at the effective address specified by ADR1, a comparison based on the first data parameter takes place. If the H3, H4, or H5, H6 parameters have also been activated, a similar comparison takes place based on the second data parameter, followed by the third data parameter. When all conditions are met, a stop occurs unless the stop has been purged by other diagnostic aids.

Messages explaining the data parameters are displayed whenever the H2, H4, or H6 control character is selected for input.

C–Check Control: Either PROCESS or HARD STOP mode may be selected. In process mode, normal operation is in effect with all error and retry functions enabled. In hard stop mode, the CPU clocks will be stopped by any error

that causes a machine-check trap. The logout and instruction retry are not taken until the clocks are restarted by activating start clocks on the service frame (a maintenance function).

I–Set Instruction Counter: The instruction counter can be set to any desired value by selecting the control character to the left of the address field and entering data in a manner similar to that described for SAR COMPARE SET.

R–Rate: The rate switch can be set either to process or to instruction step. The selection made is indicated by an arrow. The start key on the keyboard is used to initiate execution of the next instruction when instruction-step rate is in effect.

System Status Indicators

The system status indicators are displayed on the manual frame to indicate system status as defined in the following text.

System: SYS is displayed when the CPU or service meter is running; the space is blank when the meters are stopped.

Manual: MAN is displayed when the CPU is in the stopped state; the space is blank when the system is running.

Wait: WAIT is displayed when the CPU is in the wait state (bit 14 in the current PSW is 1); the space is blank when the bit is 0.

Test: TEST is displayed when any manual control is not in its normal position; the space is blank when all controls are at normal setting.

PSW: The PSW is also displayed on the manual frame. Its meaning will depend on whether the system is operating in extended control (EC) or basic control (BC) mode of operation. (*Note*: The program mask bits are not defined in either mode of operation.)

For complete PSW status information, see *IBM System/370 Principles of Operation*, GA22-7000.

Program Frame

The program frame serves as the output device for the control program and other programs operating in the Model 158; it is selected automatically upon successful completion of IPL. The program frame can also be manually selected from the manual and service frames. The image is cleared or initialized by any of the following operator functions:

System Reset
System Reset (Clear)
Load
Load (Clear)
PSW Restart.

The screen format varies according to the mode of console operation selected. When the system console is operating in printer-keyboard mode, twenty-four 80-character lines are available for system-operator messages. The twenty-fifth line is used to display the system indicators and the PSW (see Figure 11). When the system console is operating in display mode (with DIDOCS support, the screen is divided into five functional areas (Figure 12)).

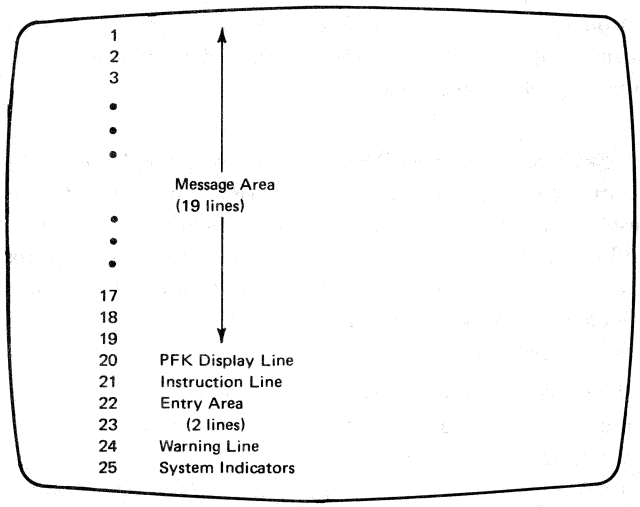


Figure 12. Program Frame Format (Display Mode)

Message Area–Lines 1 through 19: The message area is used for system messages, operator commands, and status displays. Each line contains 78 visible character positions. (The two positions preceding the first visible character position are used as system indicators and are not displayed.) The first two visible positions are used for the message number if message numbering is in effect. Positions 3, 4, and 5 contain characters that define the type of the status of the message that follows (see "Message Status Indicators"). The message text is contained in positions 6 through 78, and, if necessary, is continued on the next line beginning at position 6.

```
PROCEED REQUEST ALARM 3213  INTVN REQD  PSW=XXXXXXXX XXXXXXXX   SYS  MAN  WAIT  TEST
```

Figure 11. Program Frame Line 25 (Printer–Keyboard Mode)

Program Function Key (PFK) Display Line–Line 20: This line is blank when selector (light) pen command entry is not in use. When requested, a display with the following format appears in the PFK display line:

1 2 3 4 5 6 7 8 9 10 11 12

Only those numbers that were designated for PFK command entry functions at the time of system generation appear in the display.

Instruction Line–Line 21: This line is used for certain system messages relating to console control. Messages appearing in this line are displayed at higher intensity than other messages on the screen.

Entry Area–Lines 22 and 23: The operator uses these two lines in conjunction with the cursor and keyboard to enter commands to the system.

Warning Line–Line 24: This line contains warning messages normally requiring operator action. These messages are also displayed at high intensity.

System Indicators Line–Line 25: This line, which is not supported by DIDOCS, is used to display system indicators (see Figure 13). Selection of any position on line 25 with the light pen also serves to reset the console alarm.

Message Status Indicators: These special indicators appear in positions 3, 4, and 5 of the lines in the message area.

- A vertical line (|) in position three indicates that required action has been taken for the message or that the message can be deleted.
- A horizontal bar (–) in position 3 indicates that the message is informative and requires no operator action.
- An asterisk (*) in position 4 indicates a system message that requires action by the operator.
- A commercial at sign (@) in position 4 indicates a problem program message that requires operator action.
- A plus sign (+) in position 5 indicates a problem program WTO (Write To Operator) Message.

More detailed information on the program frame can be found in *OS/VS2 Display Consoles*, GC38-0260.

Alter/Display Frame

The alter/display frame (Figure 14) can be entered only from the manual frame when the clocks are running and the CPU is in the stopped (manual) state. The alter/display frame allows the operator to display and/or alter the following facilities within the CPU:

1) M – Real Main Storage
2) V – Virtual Main Storage
3) K – Main Storage Storage Protection Keys
4) E – Real Channel UCWs
5) U – Logical Channel UCWs
6) T – Active UCWs
7) L – CPU Local Storage
8) I – I/O UCW Local Store
9) B – I/O Buffer Local Storage
10) C – Control Registers
11) G – General Registers
12) F – Floating-point Registers
13) P – PSW
14) X – Prefix Register

The operator has the option of either keying in the function, facility, and address desired or using the light pen.

Keyboard Selection: The operator enters the function code 'A' for alter or 'D' for display. The sounding of the audible alarm and the failure of the cursor to move from the reset position (under 'K' in KEY IN/ADDRESS) indicates that an invalid function code has been entered. A valid function code must be entered before the cursor will reposition one position to the right. Next, the operator enters the facility code required. Again, if the code is invalid, the audible alarm sounds, and the cursor does not advance. When a valid code is entered, the cursor advances to the leftmost digit position in the address field. The reset value of this field is defined as 000000. If an address is required, it can be keyed in; it will be displayed left-justified as keyed. All address characters are checked for a hex value of 0 through F. Invalid characters are ignored, and the console alarm sounds.

Light Pen Selection: The operator first selects the function and then the facility desired by pressing the light pen to the appropriate control character. The cursor appears in the address field. If an address is required, it is selected from the hex input matrix at the lower right corner of the frame by pressing the pen to the required digits in turn. As each digit is selected, it appears at the cursor position in the address field.

Display: After selection is completed, the data is displayed at the center of the screen together with the address of the first element of each line if applicable. Invalid addresses are

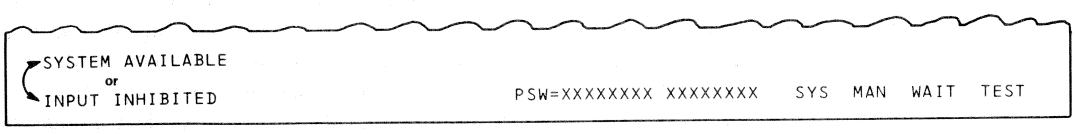


Figure 13. Program Frame Line 25 (Display Mode)

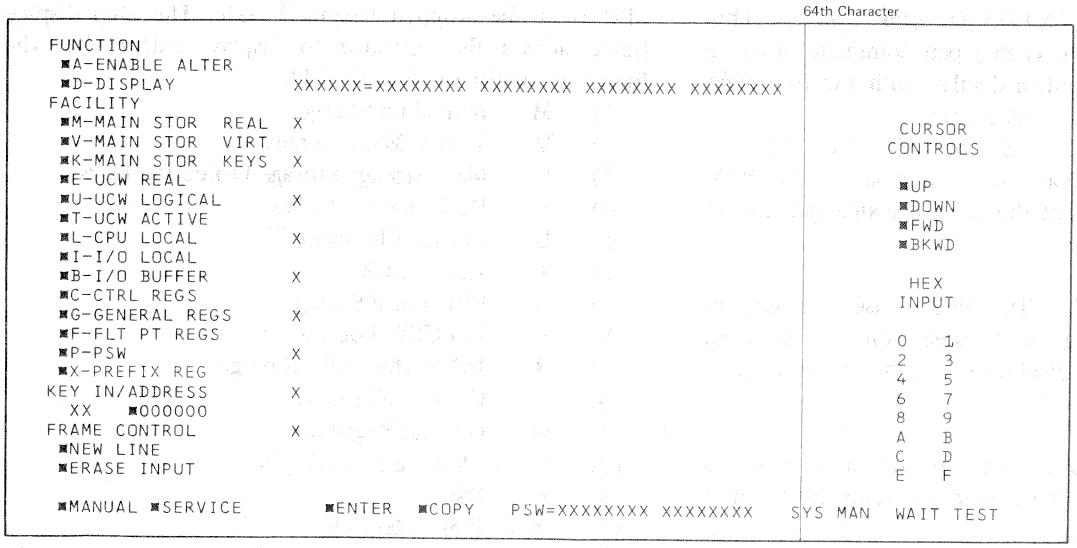


Figure 14. Alter/Display Frame

flagged in the display area by the characters '=IVA' following the invalid address

The amount of data displayed depends on the facility selected. Data is displayed on alternate lines, that is, the third, fifth, seventh, and so on. If the 'alter' function is selected, the cursor is located on the blank line under the element having the address specified at facility selection time. If 'display' was selected and the alter function is desired, 'enable alter' can now be selected, and the desired alterations can be made. Once alterations are made, 'enter' must be selected in order to store the altered data.

Should the display of more data be required, touching the light pen to the 'new line' control character (or entering special character NL (X'15') on the keyboard) shifts the data displayed on the screen up two lines, and a new line with address information appears at the bottom of the screen. If the cursor was on any line other than the top line, it maintains its relative position. If, however, it was on the top line, it will now be positioned under the first element of the new top line.

Note that, if data on the top line of the display area is to be altered, either the control character for 'enter' or the enter key must be pressed in order to store the data as altered before selecting the 'new line' functon; otherwise the data will remain unaltered.

Cursor Controls: Cursor controls are provided at the right side of the screen for light pen (selector pen) operation. For a keying operation, the space bar, the backspace key, and the cursor control keys on the right side of the keyboard can be used.

The cursor appears only on the blank lines, so that a display of the original data can be maintained as well as the changes made to it. As the data is entered, the digit appears at the cursor position under the old value, and the cursor advances. Keyed-in data is checked for a hex value of 0 through F. Invalid characters are not displayed, the cursor does not advance, and the console alarm sounds.

Enter: In order to store altered data, the enter function must be selected either from the keyboard or by use of the light pen. When this function is selected, the address appears right-justified in the address field, and the keyed-in data is transferred to main storage. The selected function and facility specifications on the frame are terminated.

Note that the enter function must be selected prior to the new line function in order to store altered data appearing on the top line of the display area.

Erase Input: Selecting this option nullifies all alterations displayed on the screen. The function, facility, and original data remain unchanged.

Copy: Copy must be selected if a printed copy of the display is desired. Pressing the copy key or selecting the copy function with the light pen is the only operation required.

HIERARCHICAL MONITORING SYSTEM

The hierarchical monitoring system (HMS) is a service aid provided by the System/370 3158-3 service processor to assist in diagnosing microcode and programming problems. The HMS is supported by an HMS frame on the console display, and all HMS controls and execution results are contained within this frame.

The HMS is capable of monitoring two facets of any program: (1) the execution of any instruction in the instruction stream and (2) the alteration of any storage content. Monitoring is done by stopping the normal execution at either of the two points listed above, at which time a number of the system facilities, such as general registers, control registers, and PSWs (status indicators) can be fetched by the HMS for testing, displaying, or storing.

The monitoring system is capable of checking system indicator status information concurrent with execution of particular reloadable control store (RCS) control words. The microcode hierarchy of monitoring and the program hierarchy of monitoring can be mixed in any HMS monitoring program.

The HMS can be used to solve significant system problems while the CPU operates normally. With or without an operating system, the HMS can monitor RCS and the application program concurrently in either a real or a virtual environment.

For a detailed account of HMS operation and programming, see *IBM System/370 Model 158 Hierarchical Monitoring System, Feature Description*, GA22-7056.

IBM 3213 CONSOLE PRINTER

The IBM 3213 Console Printer is a table-top printer that provides printed output for the Model 158 (see Figure 15). Characters are printed as a pattern of dots formed by a vertical row of seven wires striking the ribbon as the print head moves along the print line. Under program control, printing can occur at a rate of up to 85 characters per second. Character spacing is 10 per inch (25,4 mm) and up to 126 per line. Line spacing is either six lines per inch or three lines per inch depending on the setting of the line-feed select lever.

All printer-keyboard commands are recognized by the 3213 in printer-keyboard mode. In display mode, READ (0A) is not recognized (see "Printer-Keyboard Mode" under "Programmed Operations").

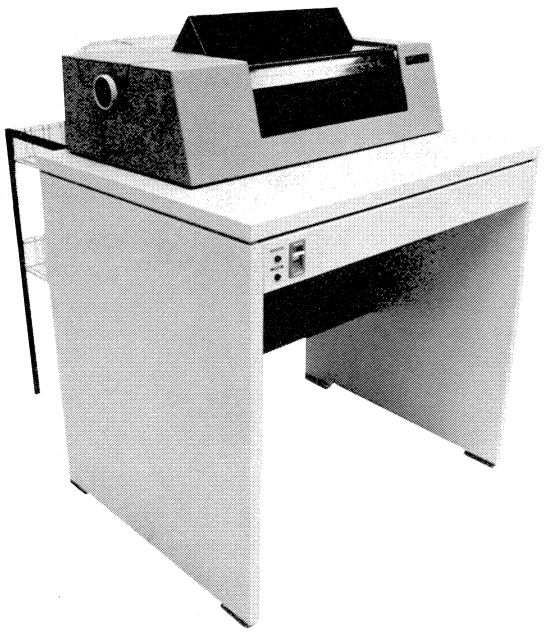


Figure 15. IBM 3213 Console Printer

All printing, carrier operations, and forms movement controls are initiated in the system. An end-of-forms switch signals the system that the end-of-form is within 4-¼ inches (107,9 mm).

Printer Controls and Indicators

External controls and indicators on the 3213 Console Printer consist of a power-on/off switch, an end-of-forms indicator, and an attention indicator.

End-of-Forms: This light comes on to indicate that an end-of-forms condition is imminent or that a cover interlock is open.

Attention: This light comes on when the power-on/off switch is off; if power is on, a missing or incorrect voltage level that requires the attention of service personnel is indicated.

Operator Controls

Paper-advance Knobs: Rotating the knob on either end of the platen moves the form for coarse vertical positioning.

Vertical Alignment Vernier: Pressing in and rotating the paper-advance knob on the right end of the platen provides fine vertical adjustment of the form to the print line.

Line-feed Select Lever: The line-feed select lever allows the operator to select either single (6 lines per inch) or double (3 lines per inch) spacing.

Copy-control Lever: The copy-control lever adjusts the clearance between the platen and the print head to accommodate the thickness of the forms used. Five positions of the lever allow for a maximum form thickness of 0.018 inch (0,46 mm).

Forms Specifications

The 3213 accepts margin-punched, continuous-form paper. Forms can be from one to six parts depending on the weight of paper and carbon used. However, for optimum feeding and stacking, forms of no more than three parts are recommended. Form width must conform to the pin-feed platen installed (see Figure 16). Form length can be from 3 to 14 inches (76.2 to 355.6 mm). For optimum stacking, 11 inches (279.4 mm) is recommended. The special ribbon cartridge has a ribbon life of at least two million characters.

Overall Width		Hole-to-Hole Width		Print Positions
Inches	Millimeters	Inches	Millimeters	Maximum
13	330.2	12-1/2	317.5	120
13-5/8	346.1	13-1/8	333.4	126

Figure 16. Available Pin-feed Platens, IBM 3213 Console Printer

The following text provides programming information for both display and printer-keyboard modes of operation. Basic to both modes of operation are the concepts of protected or unprotected data and the modified data transfer function.

Protected Data–Auto Lock: A program-controlled feature of the console allows fields to be defined as protected or unprotected. A protected field is a field that the operator may not alter in any way. If an attempt is made to enter an alphameric character when the cursor is located in a protected field, the keyboard becomes disabled by the auto lock function.

An unprotected field is one in which the operator can enter, modify, and erase alphameric data from the keyboard.

Modified Data Transfer: When it is necessary to identify data fields that have been modified, the modified data transfer (MDT) bit (bit 7 of the attribute character) is set to 1. The process is referred to as "tagging" the field.

Data in protected or unprotected fields can be tagged as having been modified. In protected fields, the tags are set under program control or by means of the light (selector) pen. In unprotected fields, modified data tags are also set by keyboard operation. These tagged fields are the only fields transferred upon execution of a read modified command.

DISPLAY MODE

In display mode, the system console from the standpoint of the operating system, performs as an IBM 3277 Display Station Model 2 with keyboard. A subset of the 3277 commands and orders is used as described in the following text.

Control Characters

Programmed control characters are the attention identifier (AID), write control character (WCC), and attribute characters.

Attention Identifier (AID)

An AID character is set in the console when one of the following events occurs:

1. The enter key is pressed (AID = 7D).
2. The cancel key is pressed (AID = 6E).
3. The selector (light) pen detects a character in a selector-pen-detectable field (AID = 7E).

The AID character causes an I/O pending condition to be set in the console, and attention status is presented to the channel. Normal program response to attention status is to issue a read modified command. The first character transferred on this command is the AID character, which identifies the cause of the attention status generation. The AID character is reset when one of the following events occurs:

1. Keyboard reset is pressed.
2. A write command is issued with the WCC keyboard reset bit set to 1.
3. An erase all unprotected command is executed.

Write Control Character (WCC)

The WCC is used in conjunction with both the write command and the erase write command to specify certain device operations. The WCC is identified by its position as the first character in a write or erase write command data stream. If an operation that does not apply to the console is specified in the WCC, the operation is ignored; no status or sense information is generated.

Bit usage in the write control character is defined in Figure 17.

Attribute Characters

Attribute characters describe the characteristics of the data fields that follow them. A formatted display is established by programmed entry of attribute characters, which are interpreted in display mode by the bit settings (refer to Figure 18).

The MDT bit in the attribute character is an exception to the general rule that an attribute character is protected from modification by keyboard action. MDT bits can be set under program control, via keyboard entry, or via selector pen detection.

Under program control, the start field order sequence in a write or erase write command data stream is used to enter an attribute character in the CRT buffer. Since the MDT bit is part of the attribute character, it can be defined as set or reset by the program. The erase all unprotected command resets the MDT bits in all unprotected fields. The reset MDT bit (bit 7) in the write control character (WCC) of a write or erase write command permits resetting of all MDT bits under program control.

Selector Pen Detection

The setting of bits in the attribute character controls the use of the selector (light) pen. A field in which selector pen

Write Control Character Bit Assignment	X	1	Not Used	Not Used	Not Used	Sound Alarm	Restore Keyboard	Reset MDT Bits
	0	1	2	3	4	5	6	7
Bit 0	Determined by the setting of bits 5, 6, and 7.							
Bit 1	Always = 1.							
Bits 2, 3, and 4	Ignored.							
Bit 5 = 1	Sounds audible alarm.							
Bit 6 = 1	Restores the keyboard and resets the attention identifier (AID) byte upon termination of the I/O command.							
Bit 7 = 1	Resets all modified data transfer (MDT) bits to 0. (The MDT bits are reset prior to the write operations so that specified modified data transfer bits can be set to 1 by the write data stream.)							

Figure 17. Write Control Character (WCC) Bit Usage

Attribute Character Bits	X	1	Unprotected/ Protected	Not Used	Intensity— Selector Pen Detection		Not Used	Modified Data Transfer
Bit Position	0	1	2	3	4	5	6	7
0	Determined by the setting of bits 2, 4, 5, and 7							
1	Always = 1							
2 = 0 = 1	Data is unprotected Data is protected							
3	Not used							
4, 5 = 0, 0 = 0, 1 = 1, 0 = 1, 1	Normal intensity field — nondetectable Normal intensity field — selector-pen-detectable High intensity field — selector-pen-detectable Nondisplay field — nondetectable							
6	Not used							
7 = 0 = 1	Field data not tagged as having been modified Field data tagged as having been modified							

Note: Bits 0 and 1 are not decoded by the console. They are transferred as received.

Figure 18. Attribute Character Bit Usage

detection is possible, and in which the consequent proper setting of the modified data transfer (MDT) bit is assured, may be protected or unprotected.

A field is selector-pen-detectable if:

1. Bits 4 and 5 of the attribute character are set to 0 and 1 (normal intensity/selector pen detectable), or 1 and 0 (high intensity/selector pen detectable), and
2. The designator (the character immediately following the attribute character) is a question mark (?), a 'greater than' symbol (>), or a blank.

If any character in a selector-pen-detectable (SPD) field is sensed by the pen, a detection is said to have occurred.

Designator

If the designator in a detected field is a question mark (?), it is changed to a 'greater than' sign (>), and the MDT bit is set to 1.

If the designator in a detected field is a 'greater than' sign (>), it is changed to a question mark (?), and the MDT bit is reset to 0.

If the designator is a blank, no change is made, the MDT bit is set to 1, and I/O pending is generated.

I/O Operation

As a result of a selector pen detection in a field containing either a question mark or a blank designator, the MDT bit is set to 1, and the field (protected or unprotected) is read back during a read modified operation. A single selector pen detection in a field with a > designator, or a double detection on the same designator character location (resulting in a two-fold change, [? to >] and [> to ?]) resets the associated MDT to 0 and prevents that field from being read back during a read modified operation.

Note: A similar operation from the keyboard (changing the alphameric value of a character, thereby causing the MDT bit to be set to 1, and then changing the character back to its original value) does not reset the MDT bit to 0.

Console Commands

In display mode, the console accepts the following commands:

Test I/O (TIO)	(00)	
Write	(01)	
Read Buffer	(02)	
No-operation	(03)	Treated as no-operation
Sense	(04)	
Erase Write	(05)	
Read Modified	(06)	
Erase All Unprotected	(07)	
Select	(0B)	Treated as no-operation

All other commands are rejected.

Note: In the following description of the write, erase write, and read buffer commands, reference is made to order sequences. A complete description of order sequences can be found following the discussion of console commands.

Test I/O (00)

The TIO command can be used to determine pending status or to test for device busy. No interruptions are generated.

Write (01)

A minimum data stream of one character, the write control character (WCC) must follow a write command. The starting buffer address for data transfers depends on the following:

1. If a WCC is followed by a set buffer address (SBA) order sequence in the write data stream, data character entry begins at the buffer address specified by the SBA order sequence.
2. If a write control character (WCC) is followed by other than an SBA order sequence, the buffer location at which data entry begins is a function of command chaining.
   a. If the write command is unchained, or chained from a select, no-op or sense command, data character entry begins at the buffer address containing the cursor.
   b. If the write command is chained from a write, erase write, erase all unprotected, read modified, or read buffer command, data character entry begins at the current buffer address.

If a write command is the first command in a chain and the data stream contains only the WCC, the command resets the buffer address to the current cursor location. The buffer address is advanced one location with each character stored. Data character storage in successive buffer locations continues until an order sequence that alters the buffer address is encountered in the data stream, or until the write command data stream is terminated.

The location of the cursor is not affected by the write command unless an insert cursor order sequence is included in the data stream.

The write command is ended when the last character in the write data stream is received. Following receipt of the last character in the write data stream, the operations specified in the WCC are executed, with the exception of the 'reset modified data tag bit' operation, which is executed prior to the receipt of the write data stream.

Read Buffer (02)

The read buffer data stream consists of the three-character read heading (AID, CUR ADR, CUR ADR) followed by the contents of all buffer locations (protected and unprotected,

attributes and nulls). The transfer starts at a specific location and continues to the end of the buffer. The beginning of each field is identified in the read-buffer data stream by a start field (SF) order code (hex '1D') followed by the attribute character associated with that field. The starting address for data transfer is a function of command chaining as follows:

1. If the read buffer command is unchained, or if it is chained from a select, no-op, test I/O, or sense command, the transfer begins at buffer location 0.
2. If the read buffer command is chained from a write, erase write, erase all unprotected, read modified, or read buffer command, transfer begins at the current buffer address.

Transfer continues until the last buffer location (1919) is reached, unless the channel count goes to 0. A typical read buffer data stream is: AID, CUR ADR, CUR ADR, SF, ATR1, DATA,..., SF, ATR2, DATA..., SF, ATR3, DATA..... etc.

No-operation Select (03 and 0B)

No-operation and select are both treated as a no-operation. Although they perform no functional operation, they may be used as a programmed switch in a command chain. Channel end and device end will be presented at initial status time unless a pending status condition or a busy condition exists.

Sense (04)

The sense command transfers one byte of sense data to the CPU in response to unit check status. The meaning of each bit is defined under "Sense Byte."

Erase Write (05)

The erase write command clears the entire storage buffer to nulls ('00'), stores any new data characters provided by the program, and performs those operations specified by the write control character (WCC). A minimum data stream of one character, the WCC, must follow the erase write command.

Following receipt of the erase write command:

1. Every character location in the buffer is set to a null ('00').
2. The buffer address is reset to 0.
3. The cursor is positioned at buffer location 0.

Data is stored in the buffer according to the contents of the erase write data stream. The WCC and all order sequences are executed in the same manner as in the write command. Order sequence operations are based on initial buffer conditions of (1) buffer address = 0, or (2) buffer is unformatted.

Operation: After accepting the erase write command, the console erases buffer location 0 to a null and inserts a cursor in that position. The console then proceeds to erase the remaining locations in the buffer to nulls, concluding the operation at location 1919. The next operation is set to start at buffer address 0. Following the erasure, the console accepts and stores the WCC. The erase write command then proceeds as a normal write command. A 'reset MDT' WCC action request is ignored on an erase write operation, since the erase operation resets all modified data transfer bits.

Read Modified (06)

When a standard read modified command is performed, all fields (protected and unprotected) in which the modified data transfer (MDT) bit has been set are read for transfer to the CPU. A three-character read heading consisting of the AID code followed by the two-character buffer address of the cursor precedes the read modified data stream. Following the read heading, the read modified data stream contains the alphameric data from each field in which the MDT bit is set, preceded by a set buffer address (SBA) order code and the two-character position in the tagged field (attribute + 1). Null characters contained in the tagged fields are suppressed; they are not included in the read modified data stream.

The buffer location at which the search begins for the first attribute character defining a field that contains modified data depends on the following considerations:

1. If the read modified command is unchained or if it is chained from a select, no-op, test I/O, or sense command, the search begins at buffer location 0.
2. If the read modified command is chained from a write, erase write, erase all unprotected, read buffer, or read modified command, the search begins at the current buffer address.

The search for modified data fields ends at the last character location (1919) in the buffer. If a modified field continues beyond the last character location and wraps around to the first, the transfer of modified data associated with that field continues to the end of the field. Any time a modified field is wrapped, it is the last field transferred, and the buffer address is set to the attribute character location of the next field. In all other cases of a normally terminating read modified operation, the buffer address is set to character address 0.

If the buffer is formatted but no fields contain modified data, only the three-character heading data is transferred. If the buffer is formatted, transfer, starting at location 0, includes the three-character heading, all 1920 characters, suppressing nulls, and excludes any SBA sequences in the data stream. The command is terminated when the last character of the last modified field is transferred, or when the channel byte count is reduced to zero. In the last case, the buffer address is left at an undefined location.

Operation: Data is transferred to the channel until the channel indicates stop, or until the last character of the last

modified field in the buffer is sent. At this time, channel end, device end ending status is sent.

When a read modified command is issued following a light pen attention, the data stream consists of the read heading (AID, CUR ADR, CUR ADR) and the *addresses* of the modified data fields (SBA, ADR, ADR). No data is transferred from the modified fields.

Short Read

When a read modified command is issued following the pressing of the CNCL key, the read modified data stream is restricted to the AID character. No cursor information or buffer data is transferred as a result of the read modified operation.

Erase All Unprotected (0F)

When the buffer is completely protected, the erase all unprotected command performs the following:

1. Restores the keyboard.
2. Resets the attention identifier (AID).
3. Moves the cursor to location 0.

When the buffer contains unprotected fields or is unformatted, the erase all unprotected (EAU) command:

1. Clears all unprotected alphameric characters to nulls.
2. Resets the MDT bits of all unprotected fields to 0.
3. Resets 'I/O pending'.
4. Repositions the cursor to the first character location in the first unprotected field in the buffer.

Erase all unprotected is executed as an immediate operation. Upon acceptance of the command, channel end is sent to the channel in the initial status byte and the CRT becomes 'busy'. Upon completion of the EAU command, an asynchronous device end is sent and the CRT becomes 'not busy'.

Order Sequences

System and channel generated order sequences may be included in the data streams of write or erase write commands only. The insert cursor order is a single-character order that is executed upon receipt of the order code. The start field, set buffer address, repeat to address, and erase unprotected to address orders are each sequences of two or more characters (see Figure 19). After the order code, the following one, two, or three characters are decoded as having the meaning specified in Figure 19. The two high-order bits of these characters are not decoded, so that if another order code is erroneously programmed (rather than the appropriate sequence character) it will not be recognized as an order, but will be assigned the value associated with its six low-order bits. The start field (SF) and set buffer address (SBA) orders are the only orders that are included in console-generated read data streams.

Start Field (SF)

The start field order is used to create a field or to modify an existing field. When included in the data stream of a write or erase write command, the SF order code (hex '1D') causes the system console to decode the character following it in the data stream as an attribute character.

The SF order occurs in the console-generated read buffer data stream as a means of identifying the attribute character associated with a transferred data field.

Set Buffer Address (SBA)

The set buffer address order sequence is used in a write or erase write command data stream to reset the buffer address to a specified character location. The SBA order code (hex '11'), when included in the data stream of a write or erase write command, causes the console to decode the two characters following in the write data stream as the specified buffer address. If an illegal address (1920–2047) is specified, the write operation is terminated immediately.

Usually, an SBA order sequence will be programmed as the first character sequence following the WCC in the write data stream of an unchained write command in order to specify a starting location. The SBA order sequence occurs in the console-generated read modified data stream as a

Order / Order Sequence	Character 1 Order Code (Hexadecimal)	Character 2	Character 3	Character 4
Start Field	1D	Attribute		
Set Buffer Address	11	Address	Address	
Insert Cursor	13	———	———	———
Repeat to Address	3C	Address	Address	Character
Erase Unprotected to Address	12	Address	Address	

Figure 19. Order Sequences

means of identifying the two characters that specify the location of the first character of a transferred modified data field.

Insert Cursor (IC)

The IC order code (hex '13'), when included in the write or erase write command data stream, causes the cursor to be relocated to the character location specified by the current storage buffer address. The buffer address is not advanced as a result of the cursor insertion. Thus, if an IC order follows a buffer character entry, the cursor is inserted in the character location immediately following the character entry. The buffer address is set at the character location containing the cursor.

Repeat to Address (RA)

The RA order sequence code (hex '3C'), when included in the write or erase write command data stream, inserts a specified alphameric or null character in all buffer locations, beginning at the current buffer address, and continuing to but not including the address specified in the order sequence. The specified address is defined by the two characters in the data stream immediately following the RA order code. The next character in the data stream specifies the character to be repeated. An RA operation may wrap from the bottom row to the top row of the display image. If the specified address is equal to the current address, the specified character is inserted in all character locations in the buffer. If an illegal address (1920–2047) is specified, the write operation is terminated at this point.

Erase Unprotected to Address (EUA)

The EUA order sequence code (hex '12'), when included in the data stream of a write or erase write command, erases all unprotected character locations in the buffer, beginning at the current buffer address, and continuing to but not including the address specified in the order sequence. The specified address is defined by the two characters in the write data stream immediately following the EUA order code. An EUA operation may wrap from the bottom row to the top row of the display image. If the specified address is equal to the current address, all unprotected character locations in the buffer are erased. If an illegal address (1920–2047) is specified, the write operation is terminated at this point.

Status and Sense Information

Status Byte

The status byte is sent to the channel during initial and ending sequences, and when the console detects an asynchronous status condition. The status byte is reset upon channel acceptance. Bit usage is as follows:

Status Byte Bit	*Definition*
P	Parity
0	Attention
1	Not used
2	Not used
3	Busy
4	Channel end
5	Device end
6	Unit check
7	Not used

Attention: The attention bit notifies the system of a need for program attention. This bit should be program-interpreted as an indication that a message is waiting to be transferred. The expected program reaction is the issuance of a read modified command. Conditions that cause the attention bit to be set alone in the status byte include:

1. The ENTER key is pressed.
2. The CNCL key is pressed.
3. A selector pen detection is made in a selector-pen-detectable field.

Busy: The busy bit is set in the status byte in the following cases:

1. When secondary status is cleared by a start I/O (SIO) instruction.
2. When a start I/O or test I/O instruction is issued between channel end and device end.

Channel End: The channel end bit is set alone in the status byte:

1. When the channel indicates the end of the data stream on a write or erase write command.
2. When the erase all unprotected (EAU) command (an immediate operation) is accepted.

Channel end is set with device end status in the following cases:

1. When a no-op or select command is accepted.
2. At the conclusion of a read buffer, read modified, or sense command; or when the channel byte count goes to zero on a read buffer or read modified command.
3. When channel end status is stacked by the channel and the operation is completed before the channel can accept status.

Channel end is set with device end and unit check status in the following cases:

1. An illegal buffer address or an incomplete order sequence is received from the channel on a write or erase write command.
2. An internal parity check is detected by the console while transferring data to or from the channel.

3. Channel end status is stacked by the channel and the remaining command operations are not completed successfully before the channel accepts status.

Device End: Device end is set with channel end and with channel end and unit check as previously stated. Device end is set alone in the status byte:

1. After completion of the WCC functions specified in a write or erase write command.
2. After completion of an erase all unprotected (EAU) command.

Unit Check: Unit check indicates to the program the existence of a condition at the system console requiring investigation. Unit check is set alone in the status byte only if the channel issues a command not recognized by the console.

Sense Byte

The sense byte is sent to the channel in response to a sense command. Bits within the sense byte are set to record unusual conditions that occur within the console. The sense byte is reset only upon the successful initiation of a start I/O instruction (condition code = 0). It is not reset by NOP, sense or select commands. Bit usage is as follows:

Sense Byte Bit	*Definition*
P	Parity
0	Command reject
1	Not used
2	Not used
3	Equipment check
4	Not used
5	Not used
6	Not used
7	Operation check

Command Reject: The command reject bit is set in the sense byte if the console receives an invalid command code.

Equipment Check: The equipment check bit is set in the sense byte if an internal parity error is detected while the console is transferring data to or from the channel.

Operation Check: The operation check bit is set in the sense byte to indicate that the console cannot execute the programmed command. This bit is set:

1. When an illegal buffer address is received with a set buffer address (SBA), repeat to address (RA), or erase unprotected to address (EUA) order in the data stream of a write or erase write command.
2. When the channel ends the write data stream before sending all of the required characters for an SBA, RA, EUA, or SF order sequence.

Interruptions

Normal Interruptions

The following text describes conditional status interruptions that may occur during normal error-free operation.

Initial Status:

1. Channel end and device end bits set in the status byte. This is a normal response to either a no-op or select command.
2. Channel end bit set alone in the status byte. This is a normal response to an erase all unprotected command.
3. All-zeros status byte. This is normal status for all other commands.

Ending Status:

1. Channel end bit set alone in the status byte. This status is sent to the channel at the end of the data stream of a write or erase write command.
2. Channel end and device end bits set in the status byte. This status is sent to the channel (1) at the end of the data stream of a read buffer, read modified, or sense command; or (2) when the channel byte count goes to zero on a read modified or read buffer command.

Asynchronous Status:

1. Attention bit set alone in the status byte. This status is sent to the channel following an attention-generating action by the operator (e.g. pressing the enter key, pressing the cancel key, light pen detection in a detectable field).
2. Device end set alone in the status byte. This status is sent to the channel only if the operations specified in the write control character (WCC) of a write or erase write command are completed.
3. Channel end and device end bits set in the status byte. This status is sent to the channel if the channel end is stacked by the channel and the operation is completed before the channel can accept status.

Error Interruptions

The following text describes error interruptions that can occur during initial status and ending sequences of command operations.

Initial Status: Unit check and command reject bits set in the sense byte. This indicates that the command decoded was invalid.

Ending Status:

1. Channel end, device end, and unit check bits together with data check bit set in the sense byte. This indicates that an internal parity check was detected while handling data on a write or erase write command.

2. Channel end, device end, and unit check bits together with operation check bit set in the sense byte. This indicates that (1) an illegal buffer address was received in the data stream of a write or erase write command, or (2) the data stream ended before providing all of the characters required for an SBA, RA, SF, or EUA order on a write or erase write command.

Asynchronous Status: No asynchronous error interruptions will occur.

PRINTER-KEYBOARD MODE

In printer-keyboard mode, the console provides the following facilities:

1. Direct data entry from the keyboard.
2. Printer and display output from the system.
3. Control of all system functions via keyboard operation.

The keyboard, console display, and printer all have the same address, which is designated by one plug card at installation time.

Commands

The console accepts the following commands in printer-keyboard mode:

Sense	(04)
Control (No-op)	(03)
Control (Alarm)	(0B)
Write (ACR)	(09)
Write (ICR)	(01)
Read	(0A)

All other commands are rejected.

Sense

The sense byte is read from control storage and is placed in the main-storage location specified by the address in the sense command. If the unit is not operational, the sense command is still executed. The intervention required bit is on. The byte count in the sense command should be 1. If the count is greater than 1, an incorrect-length (IL) indication results, provided that the SLI flag is off. Channel end and device end status bits are presented in the CSW stored by a subsequent I/O interruption (or cleared by test I/O) for the sense operation.

Control (No-op)

No-op is an immediate command that is processed regardless of whether the console printer is operational. Unit check and intervention required bits are not set when a no-op command to a nonoperational unit is executed. Channel end and device end status bits are set in the CSW stored for a start I/O initial selection that called for a no-op command (provided command chaining is not in progress).

Control (Alarm)

Control alarm is an immediate command that functions like the no-op command, except that the audible alarm in the CPU sounds and a flashing 'ALARM' indicator appears on the display when the command is executed. The audible alarm sounds whether or not the console printer is in the ready condition. The alarm is reset *only* by light pen selection.

Write (Automatic Carrier Return)

The write command is accepted by the console printer only if the following conditions are satisfied:

1. The unit is ready; that is, the forms are in place and no interlocks are open.
2. The write command has a valid format (the byte count is not 0 and the data address is valid).
3. The unit is not busy.

If the unit is not ready, condition code 1 is set and unit check status in the CSW is stored for the start I/O initiating the write command, or is stored on a subsequent I/O interruption (or test I/O) if chaining to the write command was performed.

When the print operation moves the carrier to the end of the print line, the carrier activates the end-of-line switch which automatically initiates a new-line (carrier return and line-feed) function.

When the print operation is completed and the byte count reaches 0, an automatic new line (NL) function is performed. When the end-of-line switch is operated after the last character is printed, *two* new-line functions occur, one because of the switch and the other because of the write (ACR) command.

Write (Inhibit Carrier Return)

This command is performed the same as the write (automatic carrier return) command, except that no new-line function is performed after the byte count in the write operation reaches 0, except when the fixed-margin switch is detected by hardware after the last character is printed.

Read

The read command is accepted by the console printer only if the unit is both ready and not busy.

If the unit is not ready, the unit check status bit set in the CSW is stored for the start I/O initiating the read command, or is stored on a subsequent I/O interruption (or test I/O) if chaining to the read command was performed.

If the end-of-line switch is operated by the carrier after a character is printed, a new-line function occurs but the new-line character is not sent to main storage. Proceed is off for the duration of the new-line operation.

If the enter key is pressed, the read operation is ended. If the count is not 0 and the SLI flag is off, incorrect length

(IL) is indicated in the CSW stored for the operation. The enter character bit pattern is not sent to main storage and nothing is printed as a result of an enter key operation.

If the cancel key is pressed, the unit exception status bit is set in the status byte. If the byte count is not 0 and the SLI flag is off, IL is indicated in the CSW stored for the operation. No character is sent to main storage, but an asterisk is printed and the carrier returns (NL function).

If the byte count is 0, the operation is ended the next time any key is operated. The character is not sent to main storage and nothing is printed. If any key other than the enter or cancel key is operated, IL is indicated in the CSW stored when the SLI flag bit is 0 (off).

At the end of each read operation (count equals 0, or manual end), the printer-ready condition is tested. If the printer is not ready, any chaining called for is disallowed. Unit check, channel end, device end, and intervention required bits are set as ending status in the console printer status and sense bytes.

Status and Sense Information

Status Byte

The status byte is sent to the channel when status information is required during an operation. The status byte format is:

Bit	*Definition*	*CSW Position*
0	Attention	32
1	Not used	33
2	Not used	34
3	Busy	35
4	Channel end	36
5	Device end	37
6	Unit check	38
7	Unit exception	39

Attention (Status Bit 0): This bit is set in the status byte when the request key is pressed if no other operation is in progress. If another operation is in progress, pressing the request key causes the attention status bit to be turned on after status for the other operation has been cleared (accepted by the channel).

After the attention status bit is set in the status byte:

1. If an I/O interruption for the console printer is processed, the CSW stored contains attention (bit 32).
2. If a start I/O is executed before the I/O interruption can be processed, the CSW stored for the start I/O contains attention (bit 32) plus busy (bit 35).
3. If a test I/O is executed before the I/O interruption can be processed, the CSW stored for the test I/O contains attention (bit 32).
4. If a halt I/O is executed before the I/O interruption can be processed, the CSW is not stored and the condition code is 0 (interruption pending).

Items 1, 2, and 3 clear the status at the console printer; item 4 does not clear the status.

Busy (Status Bit 3): This bit is set in the CSW (bit 35) stored as a result of execution of a start I/O only for the following conditions:

1. A program operation (other than a no-op or an alarm command) has been completed to the point at which the channel end bit has been accepted by the CPU (and an I/O interruption or test I/O instruction has been processed to store the channel end bit in a CSW), but the device end bit is now outstanding. Device end bit (CSW bit 37) accompanies the busy bit in the CSW for the start I/O, and the status at the console printer is cleared.
2. Attention status bit (resulting from a request-key operation) is outstanding for the console printer (that is, the attention status has not yet been cleared by an I/O interruption or test I/O operation). Attention bit (CSW bit 32) accompanies the busy bit in the CSW stored for the start I/O.
3. A device end bit for a not-ready-to-ready sequence is outstanding at the console printer. Device end (CSW bit 37) accompanies the busy bit in the CSW stored for the start I/O.
4. A program operation has been completed to the point at which the channel end bit has been accepted by the CPU (an I/O interruption or test I/O instruction has been processed to store the channel end bit in the CSW), but the device end bit is not yet available. The busy bit alone is presented in the CSW for the start I/O, and the console printer status is not affected.

Busy bit is stored as a result of a test I/O instruction only if test I/O is executed after the channel end bit for a command is accepted, but before the device end bit for that same command occurs.

Channel End (Status Bit 4): This bit is set for the console-printer under any of the following conditions:

1. A byte count of 0 is found in a write (automatic carrier return), write (inhibit carrier return), read, or sense command, and carrier return is completed.
2. A control no-op or control alarm command is accepted and executed during the initial selection sequence.
3. The enter key or the cancel key is pressed during a read command operation.
4. The enter key is pressed during a write command operation.
5. A sense command specifies a byte count greater than 1 and the console printer terminates the operation after one byte is transferred (normal operation).

Channel end bit alone, either held in the byte multiplexer channel or stacked at the console printer, is cleared by an I/O interruption or by test I/O, and is stored in the CSW.

Device End (Status Bit 5): This bit is set for the console printer under any of the following conditions:

1. A carrier return function ends for a write (automatic carrier return) operation.
2. The function immediately following the 0-count condition for the write (inhibit carrier return) operation is initiated.
3. The console printer accepts a control no-op or control alarm command during initial selection.
4. The sense byte is presented to and is accepted by the channel.
5. A not-ready condition in the console printer has been reset by inserting forms or by closing the cover.

Device end bit generated by or stacked at the console printer is cleared during initial selection for a start I/O only if channel end bit (as a result of the operation) has already been stored in the CSW by an I/O interruption or a test I/O. Busy bit accompanies the device end bit in the CSW stored for the start I/O. Test I/O clears any outstanding device end bit; halt I/O does not clear the device end bit.

Unit Check (Status Bit 6): This bit is set under any of the following conditions:

1. A character with even parity is sent from the keyboard to the CPU during a read command operation. Equipment check bit (sense bit 3) is also set for this condition.
2. If a parity error is detected on data during a write operation, a check condition is indicated in the same manner as for other byte multiplexer channel operations.
3. The forms switch indicates that the unit is out of paper or is in the not-ready condition, but then only:
   a. *After* a read or write (either type) command operation, or
   b. At initial selection for a read or write (either type) command, or
   c. During execution of a test I/O instruction to the console printer. Intervention required bit (sense bit 1) is also set for this condition.
4. An invalid command byte is sent to the console printer. Command reject bit (sense bit 0) is also set for this condition.
5. The printer fails to print within 2 seconds of the time it should. Equipment check bit (sense bit 3) is also set.

Unit Exception (Status Bit 7): This bit is set if the cancel key is operated *during* a read command operation. Data in the buffer is transferred to main storage. The read operation is terminated (channel end status bit is set). If the byte count is not 0 and the SLI flag is off for the read command, the incorrect-length indication (CSW bit 41) is also given during a subsequent I/O interruption or test I/O operation.

Sense Byte

The sense byte information is sent to the channel in response to a sense command. The sense byte format is:

Bit	*Definition*
0	Command reject
1	Intervention required
2	Bus-out check
3	Equipment check
4–7	Not used

Unit check status bit is set whenever one or more of the sense bits are set on.

Command Reject (Sense Bit 0): This bit is set if a command not defined for the console printer is issued.

Intervention Required (Sense Bit 1): This bit is set only for a read or write command when the unit is in a not-ready condition, the forms switch indicates an out-of-paper condition, or the cover interlock switch is open.

Bus-out Check (Sense Bit 2): This bit is set when even parity is detected on a character sent to the console printer from the CPU.

Equipment Check (Sense Bit 3): This bit is set when even parity is found on a keyboard-generated character, or when the printer fails to print within 2 seconds of the time when printing should begin.

The Model 158 offers two multiple-processor systems: the Model 158 multiprocessor (MP) system and the Model 158 attached processor (AP) system.

MULTIPROCESSOR SYSTEM

Multiprocessing permits two Model 158 systems to function with a single operating system in a shared main-storage environment, and provides shared I/O and floating storage-addressing capabilities.

The Model 158 MP system consists of two 3158 or 3158-3 Processing Units (both multiprocessor models), an IBM 3058 Multisystem Unit, and I/O devices with shared and nonshared control units. The system has shared main storage, as well as configuration, partitioning, and synchronization facilities.

Each CPU can function independently when in the uniprocessor (UP) mode.

System availability is increased through better use of resources. Required maintenance may be performed with reduced effect on the system operations. When servicing is required on the MP system, a maintenance subsystem consisting of a processor, channels, adequate storage, and I/O can be configured to perform the maintenance function. In general, this is accomplished by use of manual reconfiguration and vary offline facilities.

The CPUs of an MP system may be either 3158s or 3158-3. If two 3158-3s are used, their storage capacities can be unequal. Other combinations require equal main storage capacities. All features announced for the Model 158 are available for installation on both CPUs. Channels are dedicated to the CPUs to which they are attached; devices on the CPUs may be shared if the two-channel switch feature is installed.

The Model 158 multiprocessor system is available in eight storage capacities. Six symmetric (equal) storage configurations are as follows:

	Multiprocessors	*Units*	*Logical Units of Floating Storage Addressing*
1,024K	2 Model MP1s or M31s	4 –	256K logical units with 512K physical partitioning
2,048K	2 Model MP2s or M32s	4 –	512K logical units with 1,024K physical partitioning
3,072K	2 Model MP3s or M33s	6 –	512K logical units with 1,536K physical partitioning
4,096K	2 Model MP4s or M34s	8 –	512K logical units with 2,048K physical partitioning
6,144K	2 Model MP5s or M35s	6 –	1,024K logical units with 3,072K physical partitioning
8,192K	2 Model MP6s or M36s	8 –	1,024K logical units with 4,096K physical partitioning

Asymmetric (unequal) storage configurations can be put together with any two-unit combination of M32 (1,024K), M34 (2,048K), M35 (3,072K), and M36 (4,096K) models.

The physical dimensions of the Model 158 multiprocessor system are equivalent to two uniprocessor systems and the 3058 Multisystem Unit on which the configuration control panel is mounted (Figure 20).

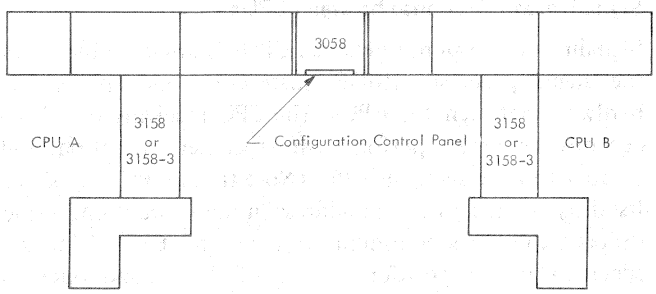


Figure 20. Example of a Model 158 Multiprocessor System (Plan View)

The following items discuss various areas in which multiprocessing differs from uniprocessing, both architecturally and in hardware implementation.

Prefixing

Each CPU in a shared-storage multiprocessing environment requires an area of storage to be used for permanently assigned locations and logout areas. Since there is only one set of storage locations in shared main storage, a means of assigning addresses to two different storage areas, one for each CPU, is necessary. The technique is called prefixing.

Prefixing in the Model 158 provides the means of assigning real addresses 0 to 4095 to any storage area, starting at an address that is a multiple of 4096. Each CPU has a private page frame allocated to it in virtual queue space for use as a permanent storage area (PSA). The real storage address for each CPU is placed in bits 8-19 of the prefix register located in the storage control unit (SCU) of that CPU. *Note: Real* addresses 0 to 4095 for each CPU are used by the nucleus initialization program and other CPU functions. The prefix register may be altered or displayed by use of the alter/display function (see "Alter/Display Frame").

Prefixing operates as follows: Whenever a CPU refers to a storage address in the range of 0 to 4095 (the high-order 12 bits, 8-19, of the effective storage address are 0's), the contents of the prefix register for that CPU replace bits

8-19 of the effective address. The new address will then point to a location in the PSA of that CPU. This is forward prefixing. When a CPU refers to an address in the 4K block that is pointed to by its prefix register (that is, an address in its own PSA), 0's are substituted for bits 8-19 of the effective address so that an address range of 0 to 4095 results. This is reverse prefixing. The prefix is applied to references made by the CPU including hardware-generated addresses such as the interval timer and PSWs. References made by a channel to channel command words (CCWs) and I/O data are not prefixed.

Signaling and Response between CPUs

Signaling and response between CPUs is provided by use of the signal processor (SIGP) instruction and appropriate hardware between the CPUs. The CPU receiving the signal decodes the order, performs the specified operation, and responds to the signaling CPU. (Note that a CPU may signal itself by inserting its own address in the instruction.) Nine orders are used for communication between CPUs. They are specified in bit positions 24-31 of the second operand address of the SIGP instruction and are encoded as follows:

Code	*Order*
(00)	(Invalid)
01	Sense
02	External call
03	Emergency signal
04	Start
05	Stop
06	Restart
07	Initial program reset
08	Program reset
09	Stop and store status
(0A-FF)	(Invalid)

Refer to *IBM System/370 Principles of Operation*, GA22-7000, for details concerning the signal processor instruction.

Shared Storage

Communication between CPUs becomes active when both CPUs have read a valid shared-storage configuration from the configuration control panel. This ensures that all references to shared storage access the most current data.

Malfunction Alert

When either CPU enters the check-stop state, a malfunction alert signal is sent to the other CPU. This signal generates a request for an external interruption. The interruption request remains pending until the interruption is taken or until the system is reset. If the interruption is taken, the address of the failing CPU is stored in locations 132-133, with an interruption code of hex 1200 stored in locations 134-135.

Time-of-Day (TOD) Clock

When the system is in MP mode, the oscillator in CPU A (hex address 0000) sends a pulse to CPU B (hex address 0001) to enable both clocks to run as one. To facilitate the programming-dependent synchronization of the time-of-day clocks, a sync pulse is transmitted each second between storage control units. This pulse can be used during synchronization to restart the TOD clock. In UP mode, the clocks run independently.

Configuration Control Panel

The configuration control panel located on the configuration control unit controls the system mode, main storage and I/O unit allocation, floating storage addressing, and system configuration. The panel contains the system mode switch, valid configuration lights, the enter configuration (ENTER CONFIG) pushbutton, a floating address switch and two storage allocation switches for each storage unit, and I/O unit allocation switches. (See Figure 21.)

System Mode Switch

The system mode switch is used to allow or disallow communication between CPUs in conjunction with the enter configuration pushbutton. If the ENTER CONFIG pushbutton is pressed when the switch is in the MP position, the signals associated with communication between CPUs can be transmitted. If the switch is set to UP, no signals can be transmitted between CPUs.

Storage Allocation Switches

Two storage allocation switches, one per CPU, are associated with each floating storage address rotary switch. Each storage allocation switch enables or disables the operation of the associated CPU with the storage unit when the ENTER CONFIG pushbutton is pressed.

Floating Storage Addressing

There is one floating storage address rotary switch per storage range assignment. Address ranges in increments of 256K, 512K, or 1,024K may be assigned, depending on the total storage capacity.

Enter Configuration (ENTER CONFIG) Pushbutton

The enter configuration pushbutton is used to enter the states of the system mode switch, the storage allocation switches, and the floating storage allocation switches into the CPUs.

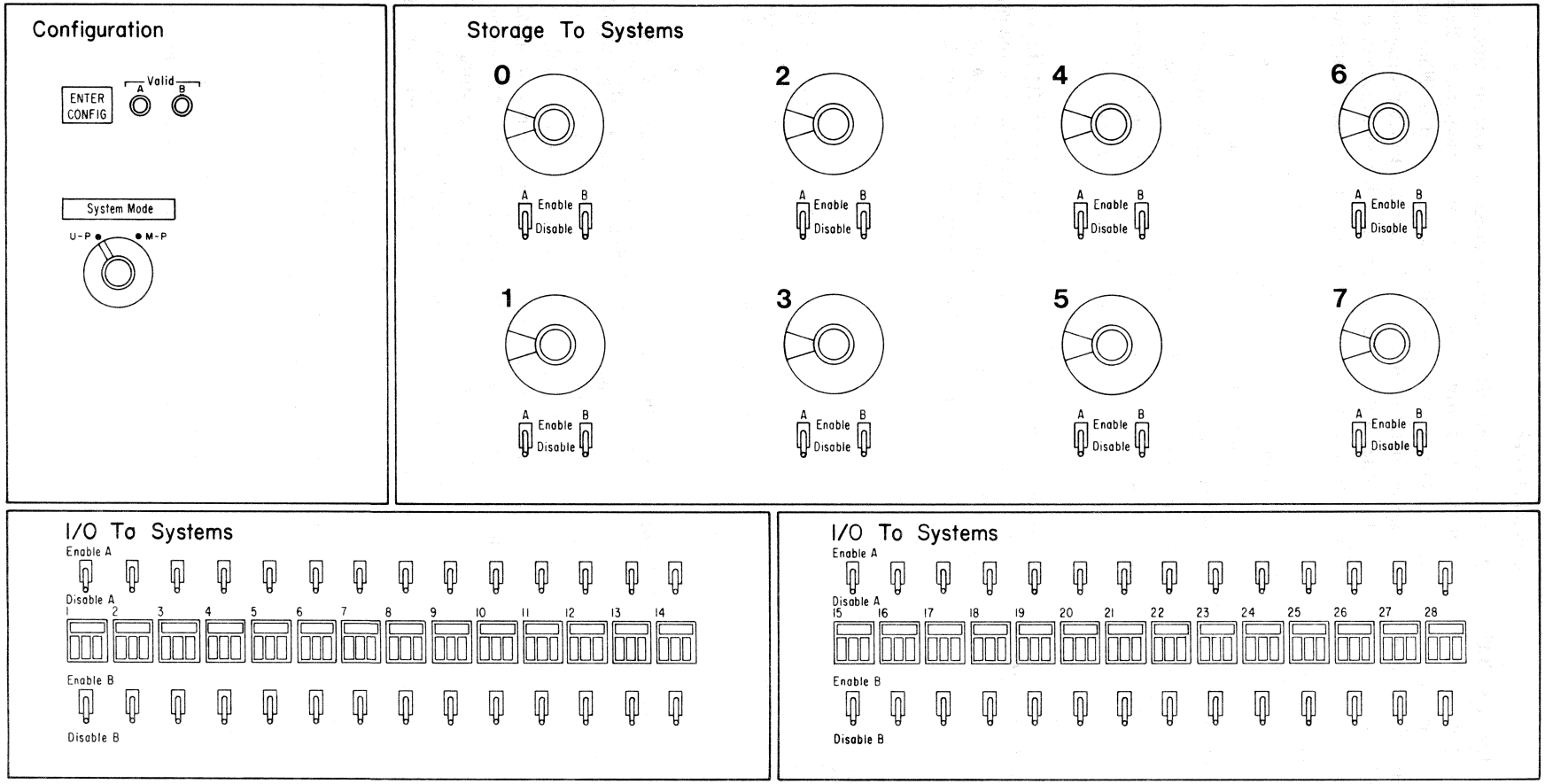


Figure 21. Model 158 MP Configuration Control Panel

Valid Configuration Indicators

Two valid configuration indicator lights, one for each CPU, are located on the configuration control panel. They indicate whether a valid system configuration has been set on the panel. Pressing the ENTER CONFIG pushbutton has no effect on a CPU whose valid indicator is off.

The indicators are turned off if:

- The system mode switch is set to MP and more than one floating storage address switch is set to one address range.
- Power is off on one CPU (that CPU's indicator will be turned off).
- A system is configured with storage but without a storage unit 0.
- The system mode switch is set to UP and a storage unit is assigned to both CPUs.
- The system mode switch is set to MP and a storage unit is assigned to one CPU but not to the other.

I/O Allocation Switches

Up to 28 pairs of I/O allocation switches are available; of these, 14 pairs are optionally available. With the two-channel switch and remote switch attachment features installed, these switches provide for the sharing of control units. They operate independently of the enter configuration pushbutton in enabling or disabling the control units for the particular CPU and channel.

Configuration Restrictions

Before configuring a CPU into the system, the CPU being brought into the running system must have had the reset/clear function invoked manually.

To vary a CPU online, a SIGP initial program reset function must be issued by the operating system before restarting the "varied-on" CPU. This must be done to synchronize the storage protection keys.

Failure to observe these restrictions will cause unpredictable results.

In addition, when performing the vary online function of a storage range, the operating system must validate that storage increment before using the new storage. Otherwise, residual errors may result in system damage. Set key instructions must be issued for the storage range varied online before using the new storage.

Manual Controls

All manual controls of the multiprocessor system function as in the uniprocessor system except system reset, clear control switch, load, and clock security switch. When the system is operating in MP mode, the signals are propagated to the other CPU. The following table summarizes the effects of reset, reset clear, load, and load clear functions:

	Effect	
Function	*Local*	*Remote*
System reset (normal)	Program reset	Program reset
System reset (clear)	System clear	Initial program reset
Load (normal)	Initial program reset	Program reset
Load (clear)	System clear	Initial program reset
PSW restart	Initial program reset	Program reset

When the system is operating in MP mode, the TOD clock is secure only when the clock security switches are secure on both CPUs. Both TOD clocks are enabled for setting if either of the clock security switches is in the enable set position.

System Control Panel Features

The system control panel (Figure 5) in a 3158-3 multiprocessing system provides additional capability, through the use of the remote/local clock switch and the remote clock indicator.

Remote/Local Clock Switch

This two-position lever switch allows main storage of the local CPU to be synchronized and timed with the clocks of the remote CPU. (The terms *local* and *remote* are relative. The local CPU, for example, is the one whose control panel is actually being operated; the remote CPU is, at that time, the other CPU.) The switch in the remote position (up) enables the remote CPU (and a segment of the local CPU's main storage, if desired) to stay online. The local CPU and console may then be powered-down for maintenance, while main storage of the local CPU remains powered-up and available to the remote CPU. In normal operation, this switch is in the local position (down).

Remote Clock Indicator

Used in conjunction with the remote/local clock switch, this indicator lights when the remote clocks are being used to time main storage of the local CPU.

ATTACHED PROCESSOR SYSTEM

The Model 158 attached processor (AP) system (Figure 22) consists of an IBM 3052 Attached Processing Unit (APU), a Model A-series 3158 or 3158-3 Processing Unit (CPU), an IBM 3056 Remote System Console, and various attached I/O devices with shared and nonshared I/O control units. The system operates under a single system-control program and can increase internal performance by 50 to 80% over that of the uniprocessor system, with identical job streams and configurations under MVS 3.7.

The 3052 APU shares the CPU's main storage. All I/O activity is performed by the host processor.

A Model A-series CPU permits attachment of the APU and its control lines to the CPU. An installed 3158 or 3158-3 may be upgraded to a Model A-series version.

The 3056 console attaches to the service processor of the APU and provides the display and keyboard for APU diagnostic and maintenance purposes. If the user wants a remote console for system use, another 3056 can be attached to the CPU.

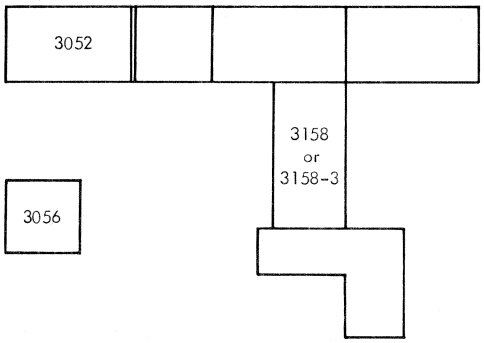


Figure 22. Example of a Model 158 Attached Processor System (Plan View)

The AP system is supported by OS/VS2 (MVS) with a selectable unit designed for the AP system. This system can be loosely coupled with other System/370s (either uniprocessor or multiprocessor) to better meet user needs. The AP system is also supported by VM/370, Release 4 or 5.

The improvement capability in the internal performance of the AP system is based on both systems having identical features, I/O facilities, and multiprogram job streams running under MVS. Each AP system's ability to achieve its throughput potential depends on the degree that the user's applications are oriented toward multiprogram job streams.

The service processors of the CPU and APU (one each) operate independently of each other. Both are designed to retain important operation information under error conditions for later evaluation, either locally or at a remote site.

The current standard and optional CPU features in a uniprocessor system apply also to the CPU in an attached processor system.

3052 Attached Processing Unit

The 3052 Attached Processing Unit (APU) provides processing capabilities similar to those of the CPU. The APU uses monolithic circuitry and contains an instruction execution unit and a 16K (16,384-byte) buffer. The buffer can satisfy many requests for processor storage, making the effective storage access time much less than processor storage cycle time. The APU is metered and can be taken offline during periods of low use or for certain maintenance activities.

Optional features for the APU include direct control, extended-precision floating point, and all the Model 158 emulator features.

The APU is not attachable to the Model 158 multiprocessor system or any other System/370.

Prefixing

Prefixing is standard in an attached processor system; it is essentially the same as in an MP system. Both the CPU and APU have prefix-value registers to enable them to assign addresses 0 to 4095 to any 4K storage area, starting with any address that is a multiple of 4,096. The 12-bit prefix value in a CPU or APU can be set by execution of the set prefix (SPX) instruction and can be inspected by the store prefix (STPX) instruction. The contents of the prefix value register are set to 0 by an initial program load, an initial program reset, and by a signal processor instruction specifying either initial CPU reset or initial program reset. The prefix value register is indicated on the CRT when the CE mode of operation is selected. Application of prefixing is described in *IBM System/370 Principles of Operation,* GA22-7000.

Signaling and Response between the CPU and APU

The signal processor (SIGP) instruction provides communication between the CPU (addressed as processor 0001) and the APU (addressed as processor 0000) in an attached processor system. Nine orders are implemented for interprocessor communication. They are specified in bit positions 24 to 31 of the second operand address of the SIGP instruction and are encoded as follows:

Code	*Order*
(00)	(Invalid)
01	Sense
02	External call
03	Emergency signal
04	Start
05	Stop
06	Restart
07	Initial program reset
08	Program reset
09	Stop and store status
(0A-FF)	(Invalid)

Details concerning the SIGP instruction can be found in *IBM System/370 Principles of Operation*, GA22-7000.

CPU and APU Addresses

The store CPU address (STAP) instruction returns an ID of 0000 when executed by the APU and returns an ID of 0001 when executed by the CPU.

Details concerning the STAP instruction can be found in *IBM System/370 Principles of Operation*, GA22-7000.

Storage Control

Main storage is located in the CPU and its use is shared with the APU. Storage references are handled on a first-come, first-served basis.

Malfunction Alert

When either processing unit (CPU or APU) enters the check-stop state, a malfunction alert signal is sent to the other processing unit. This generates a request for an external interruption. The interruption request remains pending until the interruption is taken or until the system is reset. If the interruption is taken, the address of the failing processing unit is stored in locations 132-133, with an interruption code of hex 1200 stored in locations 134-135.

Time-of-Day (TOD) Clock

In AP mode, the oscillator in the APU (hex address 0000) sends a pulse to the CPU (hex address 0001) to enable both the APU and CPU TOD clocks to run as one. To facilitate the programming-dependent synchronization of the TOD clocks, a sync pulse is transmitted each second between the CPU and APU. This pulse can be used during synchronization to restart a clock.

Power Control

Power must be on to both the CPU and APU to operate as an attached processor system, but the power needs to be on only to the CPU to operate as a uniprocessor system. EPO removes power from both the CPU and APU.

STANDARD FACILITIES

Standard facilities for the Model 158 include:

System/370 universal instruction set
- Byte-oriented operand
- Key-controlled storage protection
- Interval timer
- Time-of-day clock
- Monitoring

Translation
- Dynamic address translation
- Program event recording
- Extended control mode

CPU timer
Clock comparator
Conditional swapping
PSW key handling
Channel indirect data addressing
Multiprocessing (for M and MP models)
Error checking and correction
Instruction retry
High-speed buffer storage
Reloadable control storage
Channel retry
Command retry
Byte multiplexer channel (one)
Block multiplexer channels (two)
Fast release
Clear I/O

System/370 Universal Instruction Set

The System/370 universal instruction set is described in detail in the *IBM System/370 Principles of Operation,* GA22-7000. It contains 156 instructions and includes features such as byte-oriented operand, key-controlled storage protection, interval timer, time-of-day clock, and monitoring.

Byte-Oriented Operand

The byte-oriented-operand feature permits storage operands of most unprivileged operations to appear on any byte boundary. It does not apply to instruction addresses, nor to the operands for compare and swap (CS) and compare double and swap (CDS) instructions.

Key-Controlled Storage Protection

By matching storage keys against a key in the PSW or in the channel address word, as appropriate, the key-storage protection feature prevents the unauthorized changing or use of the contents of main storage. As many as 15 programs (with associated main storage areas) can be protected at one time.

Interval Timer

The interval timer is used by the system for measuring elapsed times of relatively short duration. The interval timer may be set to any value at any time. The interval timer runs when the CPU is in the operating state; it does not run when the rate switch in the operator section of the system console is set to the instruction-step position.

Time-of-Day (TOD) Clock

The time-of-day clock feature provides a precise measure of time suitable for accurate elapsed time measurements and time-of-day indication. The clock's binary value, updated each microsecond in bit 51, can be interrogated or set by provided instructions. The total clock cycle is approximately 142 years.

Monitoring

Monitoring, with the monitor call (MC) instruction, provides a means of selectively recording designated events in the execution of a program.

Translation

The translation feature provides for dynamic address translation, program event recording, and extended control mode.

Dynamic Address Translation

The method used to convert virtual addresses to real storage addresses is called dynamic address translation (DAT). DAT is invoked by turning on bit 5 of the PSW while the CPU is operating in extended control mode. (The user may select the extended control mode by turning on bit 12 of the PSW.) With bit 12 off, the system runs in the basic control mode, and dynamic address translation is not used. For complete information on dynamic address translation, see *IBM System/370 Principles of Operation,* GA22-7000.

Program Event Recording (PER)

PER provides for the recording of information about selected program events as the events occur. The information can be used to aid in program debugging. The following events can be recorded:

- Successful execution of a branch instruction
- Alteration of the contents of designated general registers
- Fetching of instructions from designated main storage locations
- Alteration of the content of designated main storage locations

The program has control over the conditions that are considered events for recording purposes and can specify selectively one or more events to be monitored.

Extended Control Mode

Extended control mode is specified when PSW bit 12 is set to one. This feature provides for an expanded PSW format and for a CPU mode in which certain System/370 features (such as dynamic address translation) can operate. When the system is not in extended control mode, the system is in basic control mode.

CPU Timer

The CPU timer measures elapsed CPU time, and can be used to cause an external interruption after a specified amount of time has elapsed. The CPU timer is set by the set CPU timer (SCT) instruction and inspected by the store CPU timer (STCT) instruction.

Clock Comparator

The clock comparator can be used to cause an external interruption after the TOD clock passes a time specified in the executing program. The clock comparator is set by the set clock comparator (SCKC) instruction and inspected by the store clock comparator (STCKC) instruction.

Conditional Swapping

The conditional swapping feature includes the instructions compare and swap and compare double and swap.

PSW Key Handling

The PSW key handling feature includes the instructions; set PSW key from address and insert PSW key.

Channel Indirect Data Addressing (CIDA)

Channel indirect data addressing (CIDA), a companion facility to dynamic address translation, provides assistance in translating data addresses for I/O operations. It permits a single channel command word to control the transmission of data that spans noncontiguous pages in real main storage.

Multiprocessing (M and MP Models)

Multiprocessing includes the following facilities, which permit the formation of a two-CPU multiprocessing system:

- Shared main storage
- Prefixing
- CPU signaling and response
- TOD-clock synchronization

These facilities include the following:

- Four extensions to external interruptions (external call, emergency signal, TOD-clock-sync check, and malfunction alert)
- Control-register positions for the TOD-clock-sync control bits and for the masks for the four external-interruption conditions
- Set prefix, signal processor, store CPU address, and store prefix instructions

Error Checking and Correction

Every data path in the CPU is checked for parity by byte, either directly or indirectly. The adder is checked for parity in three levels: halfsum, carry, and fullsum checks. Every data path between the CPU and main storage is also checked for parity. Error correction codes apply to data stored in, and fetched from, main storage; single and double-bit error detection and single-bit error correction are performed.

Instruction Retry

Instruction retry provides a method to recover from intermittent failures, thereby increasing system reliability. For each CPU function, a unique retry routine is entered that returns the CPU to the beginning of the function or to a point in the operation that was correctly executed and then proceeds from there.

High-Speed Buffer Storage

The high-speed buffer of the 3158 satisfies many requests for storage, making effective storage-access time shorter than processing unit storage cycle time.

Reloadable Control Storage (RCS)

Monolithic reloadable control storage (RCS), in two locations, contains microprograms that control instruction execution and channel operations. RCS is loaded from a diskette drive in the console.

Channel Retry

Channel retry is performed by machine-logic CPU retry procedures plus program-logic recovery action. Where possible, channel instructions are retried using the existing CPU retry machine logic, provided that the error occurs before the I/O command is issued to the I/O device. When the device receives the command, the channel presents a limited channel logout (LCL) to the program if an error occurs. The LCL contains information for retry of the channel instruction by programs using modified error recovery procedures. When a channel-only or channel-CPU error occurs, the entire CPU and all channels perform a

logout, and CPU retry is entered. Channels affected by the error provide a channel status word (CSW) and limited channel logout (LCL) via an interruption or a condition code 1 CSW store operation.

Command Retry

Command retry is a control-unit-initiated procedure between the channel and the control unit. No I/O interruption is required. The number of retries is device-dependent.

Byte Multiplexer Channel and Block Multiplexer Channels 1 and 2

Channels 0-2, the first byte multiplexer channel and the first two block multiplexer channels, are provided with the basic Model 158.

Fast Release

Fast release provides for early release of the processor by a channel during execution of the start I/O fast release instruction. Fast release reduces the processor delay associated with the initiation of the I/O operation.

Clear I/O (CLRIO)

Clear I/O allows the use of the clear I/O (CLRIO) instruction that causes the current operation with the addressed device to be discontinued and the state of the operation at that time to be indicated in the stored channel status word (CSW).

OPTIONAL FEATURES

The optional features for the CPU include:

- Direct control
  - System/370 extended
    - Low-address protection
    - Common segment bit
    - Invalidate page table entry instruction
    - Test protection instruction
    - Twelve MVS/SE dependent instructions
  - System/370 extended, additional
  - Channel-to-channel adapter
  - Emergency power-off control (multisystem)
  - Extended-precision floating point
  - OS/DOS compatibility
  - 1401/1440/1460 and 1410/7010 compatibility
  - 7070/7074 compatibility
  - Virtual machine assist
  - OS/VS1 ECPS (extended control program support)
  - Power warning
  - Integrated storage controls and two-channel switch for ISC
  - Staging adapter for ISC
  - Two-channel switch for ISC
  - ISC control store extension
  - Remote switch attachment
  - Register expansion
  - Processor attach
  - Expanded control store
  - 3056 remote console attachment
  - 3213 printer attachment
  - Block multiplexer channels (channels 3, 4, and 5)*
  - Second byte multiplexer channel (channel 4)*
  - Main storage options (1,024K, 1,536K, 2,048K, 3,072K, 4,096K, 5,120K, or 6,144K)*

*Refer to the configurator (Figure 1) for details on channel and main storage options.

Direct Control

The direct control feature provides two instructions, read direct and write direct, and six external interruption lines. The read and write instructions provide for the transfer of a single byte of information between an external device and the main storage of the system. Each of the six external signal lines, when active, sets up the conditions for an external interruption. Additional details are in *IBM System/360 and System/370 Direct Control and External Interruption Features, OEMI,* GA22-6845.

System/370 Extended Feature

The extended feature enhances the capabilities and performance of the Model 158 when used with the MVS/System Extensions (MVS/SE) program product. This feature includes:

- Low-address protection, whose use improves system reliability, availability, and serviceability by increasing the protection of low-address main storage (addresses 0 through 511) vital to the system control program
- The invalidate page table entry instruction and the common segment bit, for use in preserving common supervisor address translations in the TLB
- The test protection instruction that performs tests for potential protection violations without causing program interruptions for protection exceptions
- The SVC assist instruction that reduces the time needed to enter MVS, executes several frequently used supervisory services, thereby improving processor performance
- Twelve MVS/SE-dependent instructions
- The fix page instruction, six tracing instructions, and four lock-handling instructions, whose use improves processor performance
- The use of the virtual-machine extended-facility assist, also part of the extended feature, permits the preceding 12 MVS/SE-dependent instructions to be executed directly by the virtual machine without requiring program interruptions, eliminating simulation and thereby improving CPU performance.

For more detailed information, see the *IBM System/370 Extended Facility,* GA22-7072.

Channel-to-Channel Adapter

The channel-to-channel adapter feature allows the establishment of a loosely coupled multisystem via one control-unit position on the respective channels of the individual systems. Only one adapter may be installed on the Model 158; the attachment may be on either the byte or the block multiplexer channel.

Under program control, the channel-to-channel adapter can operate as two independent control units. Five commands, in addition to those supplied on System/360, are available to the programmer. Expanded checking facilities are also provided. Programs written for the System/360 adapter may be run if the additional features are not enabled.

Emergency Power-Off Control (Multisystem)

Emergency power-off control is required on only one of the processing units, normally the largest, in any installation composed of more than one cable-connected processing unit and/or cable-connected units that can be operated offline.

Optionally, from two to as many as twelve EPO switches can be installed (refer to Figure 1). The emergency power-off feature interconnects EPO switches to provide, in effect, a single EPO switch.

Extended-Precision Floating Point

Extended-precision floating point includes instructions designed to handle extended-precision (28-hex-digit) floating-point operands. Extended-precision operands may also be rounded to long-precision format, and long-precision operands may be rounded to short-precision format. For additional details, see *IBM System/370 Principles of Operation*, GA22-7000.

OS/DOS Compatibility

The OS/DOS compatibility feature allows the user to run System/360 Disk Operating System (DOS) control programs (including multiprogramming) under control of the System/360 Operating System (OS) in a multiprogramming environment. Refer to *Emulating DOS Under OS on IBM System/370*, GC26-3777.

1401/1440/1460 and 1410/7010 Compatibility

The 1401/1440/1460 and 1410/7010 integrated emulator programs allow the Model 158 to execute programs and programming systems originally written for 1400 series or 7010 systems. The compatibility feature adds special instructions and internal logic to the Model 158. The integrated emulator programs use these facilities and the available instruction sets to execute 1400-type instructions in a multiprogramming environment. Unlike stand-alone emulators, integrated emulators must share the CPU and I/O devices with the operating system. In a system with multiprogramming capability, however, the time lost waiting for a shared resource is much less (on the average) than the time lost by a stand-alone emulator waiting for its I/O operations to be completed. This reduction in system wait time increases total system throughput.

7070/7074 Compatibility

The 7070/7074 emulation allows the Model 158 to execute programs and programming systems originally written for the IBM 7070 and 7074 Systems.

Virtual Machine Assist

The virtual machine assist (VMA) feature improves performance on the Model 158 that is using virtual storage systems running under VM/370. This improvement is achieved by significantly reducing the amount of time VM/370 spends in the real supervisor state. This reduction is accomplished by emulation (instead of software simulation) of certain privileged operation codes used by the VS system supervisor. Emulation is also used for shadow page table maintenance and for SVC interruption handling.

The VMA feature may not operate concurrently with the 7070/7074 compatibility feature. The system operator must determine at IMPL time whether VMA or 7070/7074 compatibility is to be loaded.

Release 2 of VM/370 provides programming support.

For additional information about VMA, see the *IBM Virtual Machine Facility S/370: System Programming Guide,* GC20-1807.

OS/VS1 ECPS (Extended Control Program Support)

The OS/VS1 ECPS provides assist by emulation of certain supervisor functions. This feature also includes and enables virtual machine assist and extended precision floating-point functions.

Power Warning Feature

The power warning feature, on the Model 158 supported by an uninterruptible power system, permits the use of controlled shutdown and recovery procedures following power line disturbances. When utility power drops 18V ± 2% below rated voltage, the feature provides an automatic interruption to the control program. Combined with OS/VS or OS/MVT programming, the power warning feature provides support for:

- Turning on the power warning bit
- Time delay before interruption
- User intercept option
- Main storage dump/restore

The uninterruptible power system provides a power reserve to take over the task of powering either the complete system, or critical components of the system, during line disturbances including complete loss of utility-furnished power. With an uninterruptible power system supporting the complete processing system, operation may continue during power failures as long as the interruption does not exceed the buffering capacity of the uninterruptible power system. The user intercept option permits the user to program (via an exit) procedures tailored to his operation.

The CPU, all channels, and control units and devices on only one channel are powered by a partial uninterruptible power system, and use of the user intercept option may be less attractive.

The elements of the power warning feature, which is field installable, are:

1. A vendor-supplied uninterruptible power system for maintaining power to the entire or critical components of the system, and a sensor to detect power line disturbances and generate a power warning signal.
2. Hardware modifications to the Model 158 to support the power warning machine-check interruption architecture.
3. Software support for the power warning machine-check interruption handler, including the dump and restore programs and the user intercept option.

Operational Characteristics: The power warning is issued when utility voltage is reduced by 18V ± 2% for a duration greater than one-half cycle. The uninterruptible power system sensor signal remains active as long as the undervoltage condition exists, and causes the CPU to generate a soft machine-check interruption. This interruption is under the control of PSW bit 13 (Machine Check Interruption Mask), and bit 7 of control register 14 (Power Warning Sub-Mask).

If the user provides an uninterruptible power system for his complete computing system, the following events will then occur:

1. The interruption will branch to a timing routine to determine if the power disturbance is transient. The duration of this timing activity is a customer option and is limited by the uninterruptible power system capacity.
2. If the disturbance is transient, control can be returned to the machine-check handler and the system continues operation.
3. If the disturbance is determined to be non-transient, control is passed to either the dump routine or to the user intercept option.
4. If the dump routine is selected, when normal power is restored, system storage can be refreshed from the dump device. This storage information is then available to assist the user in recovery and restart procedures.
5. If the user intercept option is selected, the user is able to:
   a. Ride through the power line disturbance if it is of short duration.
   b. Assess the reserve time left in his uninterruptible power system to justify continued processing.
   c. Initiate his system quiesce procedures to terminate operations within the limits of his UPS reserves. Such procedures must be developed by the user.
   d. Transfer to the dump routine which terminates all processing and preserves the contents of storage for subsequent restart procedures.

If the user has a partial uninterruptible power system, during the processing of the power warning interruption, all channels are set to the channel interruption pending state. The channel on which the storage dump is to be performed must process the microcode-initiated channel error interruption. Interruptions are allowed from that channel only. Normal error retry procedures are used. The entire contents of storage are dumped, at which time the uninterruptible power system may be turned off to conserve reserve power. When full utility power is returned to the system, the main storage contents can be restored from the dump device.

Multiprocessing: In a tightly coupled multiprocessing system, the power warning signal is sent to both CPUs. Masking in the individual CPUs determines if the interruption is processed or held pending. In MP mode all shared storage can be dumped by either CPU, depending on which one processes the interruption.

In uniprocessor mode, each CPU must process its own interruption. Only that storage configured to a CPU can be dumped by that CPU. If both CPUs require data retention in the uniprocessor environment, they must *both* have a control unit configured that has uninterruptible power system support.

Physical Planning: Installation of either a partial or full uninterruptible power system requires a significant amount of pre-installation planning. Specialists are required to determine uninterruptible power system specifications, space requirements, cable layout, etc. Users should allow six months to a year lead time prior to the desired installation date.

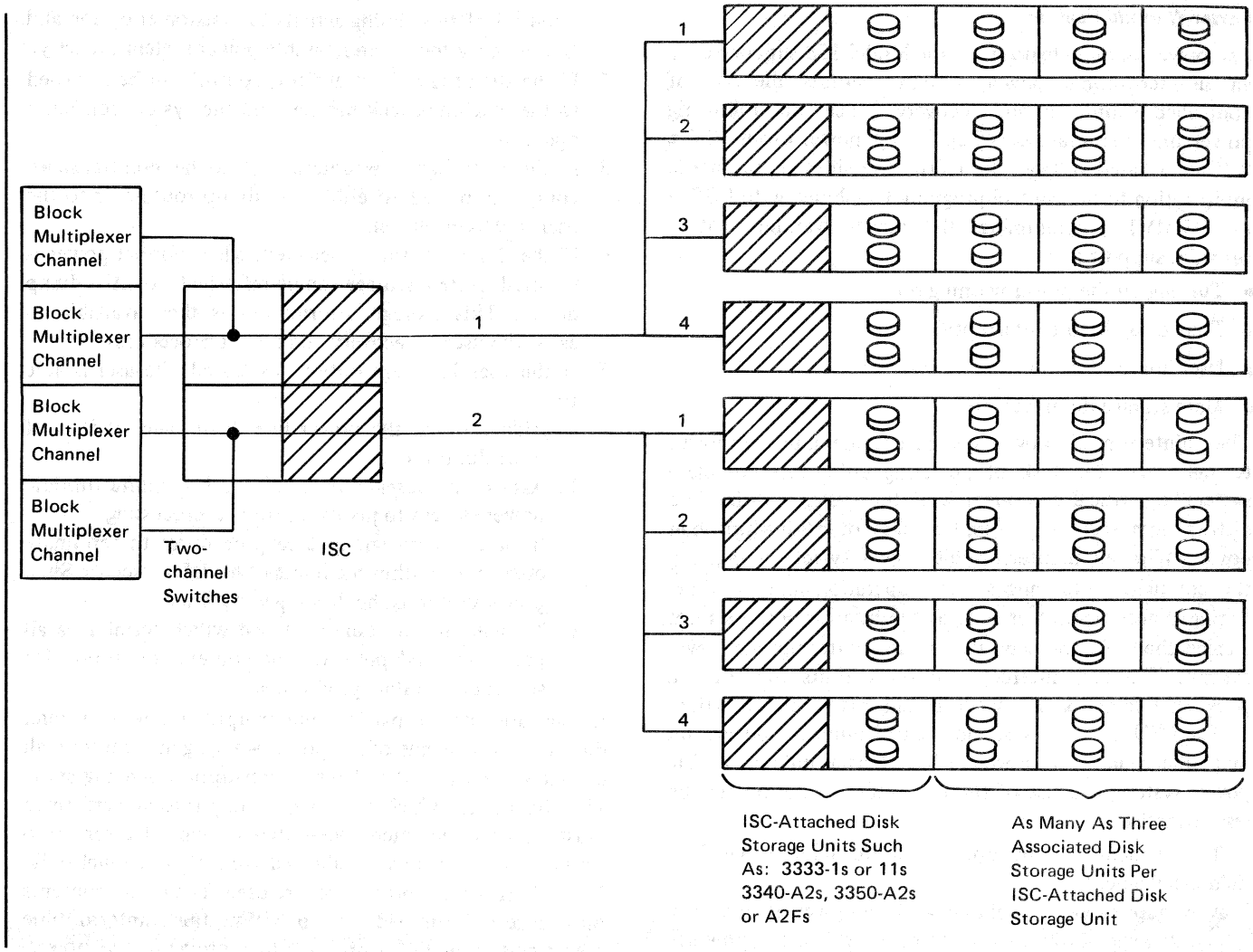


Figure 23. Integrated Storage Controls and Two-channel Switch Showing Logical Maximum Configuration

Integrated Storage Controls (ISC)

The integrated storage controls (ISC) feature provides for the attachment of 3333 Models 1 or 11, 3340 Model A2, or 3350 Models A2/A2F. Additional storage is provided by attaching 3330 modules to the 3333; 3340 Model Bs or 3344s to the 3340 Model A2 or 3350 Model B or Model C to the 3350 Model A2/A2F.

The integrated storage controls execute DASD commands; the feature is program compatible with the IBM 3830 Storage Control and the previously mentioned disk storage units in data format, channel commands, permissible command sequences, and error recovery procedures.

The ISC contains two data and control paths, each of which can attach as many as 32 disk drives (Figure 23). The two paths are logically independent, with completely overlapped operation, and each can attach to separate block multiplexer channels. There is some commonality for the two data and control paths. One kind of failure may cause both data and control paths to be inoperative.

The ISC is dependent on the 3158 Processing Unit and is clocked on the CPU meter.

Two-channel Switch

A two-channel switch feature is available that provides for attachment to an additional block multiplexer channel (Figure 23). The paths may be attached to channels on either the same of different CPUs. Channel switching and device reservation are controlled by the channel program. Two special commands are associated with two-channel switch operation: device reserve and device release. Individual drives attached to the ISC may be reserved for the exclusive use of either of the two channels attached to the given ISC path.

Staging Adapter for ISC

This optional feature for the ISC permits expansion of the addressing capability of each ISC path to a maximum of 64 unique addresses. The adapter allows direct attachment of the IBM 3850 Mass Storage System to a Model 158, and provides the same functions as the IBM 3830 Storage Control Model 3. The expanded capability that the adapter provides is independent of the number of disk storage drives attached to the ISC. Attachment of the staging adapter requires attachment of the ISC control store extension, and precludes attachment of 3340 Disk Storage drives to the ISC.

Two-channel Switch for ISC

The two-channel switch provides each of the two paths of an ISC with the capability of attaching to a second channel. The channels may be on the same CPU or different CPUs. Switching is under program control, and each path of the ISC can be dedicated to a single channel.

ISC Control Store Extension

The ISC control store extension provides 2K bytes additional control store for microprogram use on the ISC.

Remote Switch Attachment

The remote switch attachment provides the capability to attach the two-channel switch for the ISC.

Register Expansion

Register expansion provides four additional registers for microprogram use on the ISC.

Processor Attach

The processor attach feature provides storage protection for the main storage of one CPU when used by another CPU or by an attached processing unit, as in a multiprocessor or attached processor system.

In a multiprocessor system, each CPU must contain sufficent storage protection for all the main storage of the system. Therefore, in addition to storage protection for its own main storage, each CPU must have storage protection for its use of the main storage of the other CPU. Storage protection for the first 1,024K (1,048,576 bytes) of main storage of the other CPU is included in each CPU. Each processor attach feature provides a CPU with storage protection for an additional 1,024K of main storage of the other CPU. In a multiprocessor system having more than 1,024K of main storage per CPU, one processor attach feature is required in each CPU for each additional 512K or 1,024K increment of main storage of the other CPU. For example, if CPU A has 4,096K (4,194,304 bytes) of storage and CPU B has 3,072K (3,145,728 bytes) of storage, then CPU A requires two processor attach features and CPU B requires three.

In an attached processor system, the situation is similar. The attached processing unit (which has no main storage) must contain sufficient storage protection for all of the CPU's main storage. The attached processing unit is provided with storage protection for the first 1,024K of main storage of the CPU, and one processor attach feature is required for the attached processing unit for each additional 512K or 1,024K increment of main storage of the CPU. For example, if the CPU has 3,072K of main storage, the attached processing unit would require two processor attach features.

Facilities

The integrated storage controls provide or support the following standard facilities:

- Command Retry
- Multiple Requesting
- Multiple Track Operation
- Record Overflow
- End-of-File

Command Retry: Command retry is a channel-storage control procedure that causes an improperly executed command in a channel program to be automatically retried. The reexecution does not cause an I/O interruption, and programmed error-recovery procedures are not required.

Multiple Requesting: Use of block multiplexer channels and disk drives with rotational position-sensing capabilities allows the ISC and attached disk drives to disconnect from the channel during mechanical delays resulting from execution of arm-positioning seek sector or set sector commands. Reconnection is attempted when the access mechanism is positioned at the desired track, or when the specified rotational position has been reached.

During the time the channel and the ISC are disconnected, the CPU is free to initiate I/O operations on other drives attached to the ISC even though the disconnected channel program is not completed. Thus, separate channel programs may be operating simultaneously on each drive attached to the storage control.

Multiple Track (MT) Operation: On all search and most read commands, the ISC can automatically select the next sequentially numbered head on a disk drive. This eliminates the need for seek head commands in a chain of read or search commands.

Record Overflow: The record overflow function provides a means of processing logical records that exceed the capacity of a track. When the system is using overflow records, the cylinder boundary is the factor limiting the size of the record.

A special channel command (Write Special Count, Key, and Data) is used to format the disk pack for record overflow operation.

End-of-File: An end-of-file record, used to define the end of a logical group of records, is written by executing a Write Count, Key, and Data command with a data length of zero. Execution of this command causes the ISC to direct the addressed drive to write a data area consisting of one byte of zeros.

When the end-of-file record is processed, detection of the zero data length causes unit exception status to be generated.

Statistical Usage/Error Recording

The ISC maintains a statistical data record of usage and error information for each attached logical device. The usage information provides an accumulated count of the total number of access motions, and the total number of seek errors, correctable data errors, and uncorrectable data errors that were recovered by the ISC retry procedure. Also included in the error information is the total number of command and data overrun conditions that were retried by the ISC.

The usage/error information is sent to the system logout area periodically. The transfer takes place on the next start I/O issued to the device having outstanding usage/error information. Each of the usage/error counters is reset to zero after the counter information is transferred to the channel.

Storage Control Diagnostics

To provide maximum facility availability, the ISC can execute diagnostic tests on a drive concurrently with normal system operation on the remaining disk drives. This mode of operation allows servicing personnel to diagnose and repair most disk drive failures while the facility continues to operate other attached drives. The ISC provides a transient block of 512 bytes (128 words) of control storage to allow temporary residence for a specific diagnostic test.

The transient area is loaded by the system under control of the online test executive program (OLTEP). A special command (diagnostic write) loads a selected test into control storage and instructs the storage control to execute the test. This loading and execution may also be initiated from the service frame.

After the test, error-message information or test results are transferred from the ISC to main storage by a read diagnostic status 1 command. If the service frame is used, the test results are displayed on the service frame indicators.

Configuration Control

Operator-accessible switches are provided for configuration control of each ISC data and control path (see Figure 24). The ISC can operate with a given channel only when the respective interface switch is set to ON. The multitag switches determine how the device end (generated by the drive in a not-ready-to-ready sequence) is provided to the channel.

When the multitag switch is set to ON, a disk drive is available to a channel after the channel clears the device end generated by the drive on a not-ready-to-ready sequence. Before any other channel can use the disk drive, it must also accept the not-ready-to-ready sequence device end.

When the multitag switch is set to OFF, a disk drive is made available to all channels after one of the channels clears the device end generated by the drive in a not-ready-to-ready sequence.

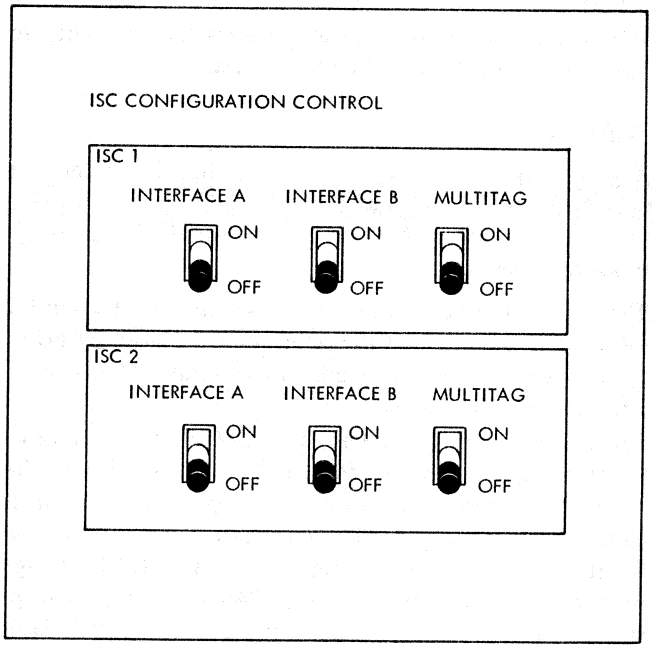


Figure 24. Configuration Control Switches with Two-channel Switch Feature

Input/Output Operations

The following text contains a general description of I/O operations related to the ISC and its attached disk storage units. For detailed information regarding the central processing unit and channel program control of I/O operations, refer to *IBM System/370 Principles of Operation*, GA22-7000.

Unit Selection and Device Addressing: The I/O address of each ISC data and control path and its attached drives is designated by an eight-bit binary number in an I/O instruction. These addresses consist of three parts: (1) The ISC data and control path address (determined by service personnel at installation time) in bits 0, 1, 2, and 3; (2) the address of the 3333 Disk Storage and Control specified in bit 4; and (3) the addresses of the attached 3330s specified in bits 5, 6, and 7. With 32 drives installed on each path, bits 0, 1, and 2 are used to designate the ISC data and control path address. The 3333 Disk Storage and Control address is specified in bits 3 and 4; bits 5, 6, and 7 retain their same function.

The ISC accepts any drive address from 000 to 111. If the specified drive is either offline or not attached, the attempted operation is terminated with unit check status. Multiple responses to an address owing to duplicate logical address plugs or hardware failures also cause the operation to be terminated.

Channel Commands: The command set used to perform operations with the ISC is identical to that used with the 3330 Disk Storage and 3830 Storage Control Model 1.

Refer to *Reference Manual for IBM Integrated Storage Control*, GA26-1620, for a description of commands and sense data.

Appendix A. EBCDIC Chart

Bits 4 5 6 7	Hex 1	00				01				10				11			
← Bits 0, 1		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
← Bits 2, 3		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
← Hex 0																	
0 0 0 0	0					SP	&	-									0
0 0 0 1	1							/		a	j			A	J		1
0 0 1 0	2									b	k	s		B	K	S	2
0 0 1 1	3									c	l	t		C	L	T	3
0 1 0 0	4									d	m	u		D	M	U	4
0 1 0 1	5		NL							e	n	v		E	N	V	5
0 1 1 0	6									f	o	w		F	O	W	6
0 1 1 1	7									g	p	x		G	P	X	7
1 0 0 0	8									h	q	y		H	Q	Y	8
1 0 0 1	9		EM							i	r	z		I	R	Z	9
1 0 1 0	A					¢	!		:								
1 0 1 1	B					.	$	,	#								
1 1 0 0	C		DUP			<	*	%	@								
1 1 0 1	D					(	)	—	'								
1 1 1 0	E		FM			+	;	>	=								
1 1 1 1	F					|	¬	?	"								

Table 1. I/O Chart–EBCDIC

These timing formulas show, in microseconds, the approximate time required to execute instructions in the System/370 Model 158. Because channel response time varies widely with the type of I/O device used, the times for the I/O instructions are not included. No CPU degradation due to channel interference is assumed.

These formulas are subject to change due to subsequent engineering change activity.

TIMING ASSUMPTIONS

The following assumptions were used in developing the instruction timing formulas:

General

1. No delay occurs during instruction execution because of conflicts caused by I/O or other asynchronous operation.
2. No program interruptions occur during the execution of the instruction (including overflow, underflow, and significance).
3. The instruction fetch time is overlapped with the execution of the previous instruction if the instruction buffer is not empty.
4. All operands and instructions are in the buffer storage 90 percent of the time and in the backing storage 10 percent of the time, unless otherwise assumed.
5. All data fetches are on appropriate boundaries; that is, the operands are on integral multiples of operand length. No boundary alignment is required. If boundary alignment should be required, the instruction times will be significantly longer.
6. The interval timer adds an additional 795 microseconds to the instruction times every 1-second of elapsed time.
7. The CPU timer adds an additional 4.5 microseconds to the instruction times every 1-second of elapsed time.
8. Each instruction time is increased by its share (which varies with the instruction type) of the time required to fill the instruction buffering registers if they should become drained during instruction processing.
9. The times do not include the local execution mode of the OS/DOS compatibility feature.
10. The instruction times do not include additional translation time for relocate mode. Translations are contained in the translation look aside buffer (TLB) 100 percent of the time.

The time in microseconds that is added to the CPU select time when translation is not in the TLB is shown in the table below:

		Segment Not in Latch	
	Segment In Latch	*Segment In Buffer*	*Segment In Backing Storage*
Page in Buffer Storage	0.690	1.150	1.725
Page in Backing Storage	1.265	1.725	2.300

11. All RX-type instructions are 0.115 microsecond faster if either the base or the index register is zero.
12. In a test and set (TS) operation, the bit tested is a 1 in 50 percent of the cases.
13. Rounding occurs after a rounding operation 50 percent of the time.

SS-Format Instructions

1. Operand 1 and operand 2 are both in the buffer 75 percent of the time. Operand 1 or 2 is in the buffer 20 percent of the time. Both operand 1 and operand 2 are not in the buffer 5 percent of the time.
2. The operands of the VFL operations do not overlap.
3. For the move character long (MVCL) instruction, 50 percent of the data fetches and all of the data stores are to main storage.
4. For the pack (PACK) instruction, the length of operand 2 is two times the length of operand 1. The length does not include the sign byte.
5. For the unpack (UNPK) instruction, the length of operand 1 is two times the length of operand 2. The length does not include the sign byte.
6. For decimal add (AP), subtract (SP), and zero and add (ZAP) operations, the length of operand 1 is greater than or equal to the length of operand 2. No recomplementation occurs for the AP and SP operations.
7. The multiply decimal and divide decimal instructions are data dependent and the calculated instruction times will vary by a significant amount for different data combinations.
8. In edit (ED) and edit and mark (EDMK) operations, 75 percent of the pattern bytes are digit-select.

Floating-Point Instructions

1. All floating-point numbers are already in normalized form.
2. All floating-point instruction times are data dependent; they depend on the number of hexadecimal digits that are preshifted due to exponent difference and postshifted for normalization and fraction overflow, as well as the number of times that recomplementation of a result occurs, and any sign difference. The times are given to one decimal point for these instructions to show that the time is data dependent and will vary by a significant amount. The times given include the following weighted averages for the listed variables (recomplementation is not included):

Add and Subtract Instructions

Time (Microseconds)	*Operation*
0.23	Preshift of long operands
0.16	Preshift of short operands
1.35	Preshift of extended operands
0.27	Postshift of normalized long operands
0.24	Postshift of normalized short operands
0.55	Postshift of extended operands
0.06	Sign difference

Compare Instructions

Time (Microseconds)	*Operation*
0.23	Preshift of long operands
0.16	Preshift of short operands
0.12	Sign comparison

Multiply and Divide Instructions

Time (Microseconds)	*Operation*
0.07	Postshift of multiply long operands
0.12	Postshift of multiply short operands
0.12	Postshift of multiply extended operands
0.23	Postshift of divide long operands
0.06	Postshift of divide short operands

3. The times required for preshifts and postshifts are:

Add and Subtract Instructions

Time (Microseconds)	*Operation*
0.230	Each preshift of long or short operands
1.150	First preshift of extended operands
0.805	Subsequent preshifts of extended operands
1.035	First postnormalization of long operands
0.690	Each subsequent postshift of long operands
0.920	First postnormalization of short operands
0.805	Each subsequent postshift of short operands
0.345	Fraction overflow for long or short operands
1.265	First postnormalization of extended operands
0.920	Subsequent postshifts of extended operands
1.495	Fraction overflow for extended operands

Compare Instructions

Time (Microseconds)	*Operation*
0.230	Each preshift of long or short operands

Multiply and Divide Instructions

Time (Microseconds)	*Operation*
0.345	Each postshift of multiply long operands
0.575	Each postshift of multiply short operands
0.575	Each postshift of extended operands
0.460	Each postshift of divide long operands
0.115	Each postshift of divide short operands

Multiprocessing

1. All the timing assumptions for uniprocessor models apply as well for the multiprocessor models if in UP mode and assigned its own storage.
2. In UP mode, uniprocessor-model instruction times apply when each CPU is assigned its main storage.

For statements 3 through 6, assume that all main storage is on-line and that both CPUs are running.

3. In UP mode, when one CPU is assigned only the main storage of the other CPU, storage instruction times can increase as much as 0.230 microsecond per storage reference.
4. In UP mode, when one CPU is assigned only one-half its main storage, and the other half is assigned to the other CPU, storage instruction times increase as much as 0.400 microsecond per storage reference.
5. In UP mode, when one CPU is assigned all its main storage and one-half the main storage of the other CPU, storage instruction times increase as much as 0.315 microsecond per storage reference.
6. In MP mode, when both CPUs are assigned all the available main storage, storage instruction times increase as much as 0.515 microsecond per storage reference. Main storage result-storing instructions increase as much as 0.790 microsecond per reference. Also, the ISK, SSK, and RRB times can increase as much as 1.495 microseconds.

Attached Processing

Statements 2, 4, and 5 under multiprocessing do not apply to the attached processing unit in an attached processing system, but can apply to the CPU. All other multiprocessing timing assumptions apply to both sides of an attached processor machine.

LEGEND FOR TIMING FORMULAS

B = Number of bytes processed. The terms B/8 and B/256 must be rounded to the next whole number.

BT = 1 If the branch is taken.
= 0 Otherwise.

CC = 1 If an unsuccessful branch is caused only by the branch-to register being zero.
= 0 Otherwise.

CM = 0 If the fields are equal
= 1 Otherwise.

C1 = 1 If operand 1 crosses a doubleword boundary.
= 0 Otherwise.

DL = Number of decimal digit positions to be shifted left.

DR = Number of decimal digit positions to be shifted right.

DW = Number of doublewords processed, not rounded.

DW1 = Number of doublewords in operand 1, not rounded.

DW2 = Number of doublewords in operand 2, not rounded.

E = Time for the subject instruction, which is executed by the 'execute' instruction. The instruction time is decremented by 0.115 microsecond if no modification of the instruction is required.

EC = 1 For EC mode.
= 0 Otherwise.

F0 = 1 If the fraction is zero.
= 0 Otherwise.

I = 1 If program interrupt is taken.
= 0 Otherwise.

I2 = Contents of the I2 field expressed as a decimal number 0–15.

M = 1 If the mask is not all zeros.
= 0 Otherwise.

MB = Number of I-bits in the mask.

N = Total number of bytes in operand 1 for instructions with a single length field. The terms N/4, N/8 and N/16 must be rounded to the next whole number.

NRD = 1 If no rounding is required.
= 0 Otherwise.

N0 = Number of high-order hex zeros in operand 1.

N1 = Total number of bytes in the first operand (destination). The terms N1/2, N1/4 and N1/8 must be rounded to the next whole number.

N2 = Total number of bytes in the second operand (source). The terms N2/4 and N2/8 must be rounded to the next whole number.

N3 = -1 If only one byte is to be processed.
= 0 Otherwise.

N8 = 1 If left shift is required.
= 1 If N1 is equal to or greater than two times the shift amount.
= 1 If N1+1 is equal to or greater than two times the shift amount and the I3 field is not equal to zero.
= 0 Otherwise.

N16 = 1 If N1 is equal to 2 or 16.
= 0 Otherwise.

N21 = 0 If N2 is greater than N1.
= 0.75 If N1 is equal to N2.
= 1 Otherwise.

PNM = 1 If postnormalization is required.
= 0 Otherwise.

P1 = 0 If operand 1 is positive.
= 1 Otherwise.

P2 = 0 If operand 2 is positive.
= 1 Otherwise.

P3 = 1 If the signs of operand 1 and 2 are different.
= 0 Otherwise.

QWB = Number of quadword boundaries crossed.

R	=	Number of registers to be loaded or stored. The term R/4 must be rounded to the next whole number.
R0	=	0 If operand 2 points to an address on a doubleword boundary.
	=	0 If an odd number of registers is to be loaded or stored.
	=	1 Otherwise.
R1	=	0 If an even number of registers is to be loaded or stored.
	=	1 Otherwise.
R2	=	0 If the number of registers to be loaded or stored is greater than 2.
	=	1 Otherwise.
S	=	1 If no 4-bit shifts are taken.
	=	0 Otherwise.
S0	=	0 If no 1-bit shifts are taken.
	=	1 Otherwise.
S1	=	Number of 1-bit shifts remaining after completion of the 4-bit shifts.
S3	=	1 If more than eight 4-bit shifts are to be taken.
	=	0 Otherwise.
S5	=	1 If shift is greater than three.
	=	0 Otherwise.
S6	=	1 If two 1-bit shifts are taken.
	=	0 Otherwise.
S7	=	1 If two or more 4-bit shifts are taken.
	=	0 Otherwise.
S9	=	1 If three 1-bit shifts are taken.
	=	0 Otherwise.

SYSTEM/370 MODEL 158 INSTRUCTION TIMINGS

Instruction	*Instruction Format*	*Op Code*	*Mnemonic*	*Time (Microseconds)*
Add	RR	1A	AR	00.495
Add	RX	5A	A	00.933
Add Decimal	SS	FA	AP	02.315+00.115(N1)+00.430(N1/4) +00.430(N2/4) + 00.660 (C1)
Add Halfword	RX	4A	AH	01.163 + 00.230 (P2)
Add Logical	RR	1E	ALR	00.380
Add Logical	RX	5E	AL	00.703
Add Normalized (Extended)	RR	36	AXR	05.0
Add Normalized (Long)	RR	2A	ADR	02.2
Add Normalized (Long)	RX	6A	AD	02.5
Add Normalized (Short)	RR	3A	AER	02.0
Add Normalized (Short)	RX	7A	AE	02.4
Add Unnormalized (Long)	RR	2E	AWR	02.0
Add Unnormalized (Long)	RX	6E	AW	02.2
Add Unnormalized (Short)	RR	3E	AUR	01.8
Add Unnormalized (Short)	RX	7E	AU	02.1
AND	RR	14	NR	00.840
AND	RX	54	N	01.163
AND (Immediate)	SI	94	NI	00.829
AND (Characters)	SS	D4	NC	00.406+00.115(N)+00.702(N/4)+00.593(N/8)
Branch and Link	RR	05	BALR	00.725 + 00.288(BT)
Branch and Link	RX	45	BAL	01.163
Branch on Condition	RR	07	BCR	00.380+00.288(BT)+00.345(CC)
Branch on Condition	RX	47	BC	00.530+00.288(BT)
Branch on Count	RR	06	BCTR	00.380+00.403(BT)
Branch on Count	RX	46	BCT	00.645+00.288(BT)

Branch on Index High	RS	86	BXH	00.875+00.518(BT)
Branch on Index Low or Equal	RS	87	BXLE	00.990+00.403(BT)
Compare	RR	19	CR	00.380
Compare	RX	59	C	00.703
Compare and Swap	RS	BA	CS	01.163+00.115(CM)
Compare Decimal	SS	F9	CP CP CP	02.315+00.115(B)+00.430(N1/4) +00.345(C1) + 00.430(N2/4) +01.610(1-N21)-00.920(P3)(1-N21)
Compare Double and Swap	RS	BB	CDS	01.393+00.403(CM)
Compare Halfword	RX	49	CH	00.933+00.230(P2)
Compare Logical	RR	15	CLR	00.380
Compare Logical	RX	55	CL	00.703
Compare Logical	SI	95	CLI	00.588
Compare Logical	SS	D5	CLC	00.785+00.115(B) +00.702(DW) +00.914(B/8)
Compare Logical Long	RR	0F	CLCL	00.725+00.368(B) +00.529(B/8) +04.037(B/256)
Compare Logical Characters Under Mask	RS	BD	CLM	00.703+01.035(M)+00.115(MB)
Compare (Long)	RR	29	CDR	02.3
Compare (Long)	RX	69	CD	02.5
Compare (Short)	RR	39	CER	02.1
Compare (Short)	RX	79	CE	02.4
Convert to Binary	RX	4F	CVB	02.198
Convert to Decimal	RX	4E	CVD	02.370
Divide	RR	1D	DR	09.350
Divide	RX	5D	D	09.903
Divide Decimal	SS	FD	DP	06.535+02.875(N1)-03.450(N1/4) -04.600(N2)+02.875(N2)(N1-N2)
Divide (Long)	RR	2D	DDR	23.2
Divide (Long)	RX	6D	DD	23.3

Divide (Short)	RR	3D	DER	08.6
Divide (Short)	RX	7D	DE	08.9
Edit	SS	DE	ED	01.130+01.668(N)+02.128(DW) -01.323(C1)
Edit and Mark	SS	DF	EDMK	01.130+01.668(N)+02.128(DW) -01.323(C1)
Exclusive OR	RR	17	XR	00.840
Exclusive OR	RX	57	X	01.163
Exclusive OR (Immediate)	SI	97	XI	00.829
Exclusive OR	SS	D7	XC	00.406+00.115(N)+00.702(N/4) +00.593(N/8)
Execute	RX	44	EX	01.623 + E
Halt Device	SI	9E01	HDV	Indeterminate
Halt I/O	SI	9E00	HIO	Indeterminate
Halve (Long)	RR	24	HDR	01.070 + 00.690(PNM)
Halve (Short)	RR	34	HER	00.840 + 00.575(PNM)
Insert Character	RX	43	IC	00.818
Insert Characters Under Mask	RS	BF	ICM	00.703+01.035(M) +00.115(MB)
Insert Program Key	SI	B20B	IPK	01.565
Insert Storage Key	RR	09	ISK	01.185
Load	RR	18	LR	00.380
Load	RX	58	L	00.588
Load Address	RX	41	LA	00.530
Load and Test	RR	12	LTR	00.380
Load and Test (Long)	RR	22	LTDR	00.610
Load and Test (Short)	RR	32	LTER	00.610
Load Complement	RR	13	LCR	00.495
Load Complement (Long)	RR	23	LCDR	00.840+00.115(F0)
Load Complement (Short)	RR	33	LCER	00.840+00.115(F0)

Load Control	RS	B7	LCTL	11.915+00.483(R) -01.955(R2) +00.345(EC)
Load (Long)	RR	28	LDR	00.495
Load (Long)	RX	68	LD	00.703
Load Halfword	RX	48	LH	00.933+00.230(P2)
Load Multiple	RS	98	LM	01.070+00.207(R)+00.023(R1) +00.115(R0)
Load Negative	RR	11	LNR	00.380
Load Negative (Long)	RR	21	LNDR	00.610+00.115(F0)
Load Negative (Short)	RR	31	LNER	00.610+00.115(F0)
Load Positive	RR	10	LPR	00.380+00.115(P2)
Load Positive (Long)	RR	20	LPDR	00.610+00.115(F0)
Load Positive (Short)	RR	30	LPER	00.610+00.115(F0)
Load PSW	SI	82	LPSW	03.072
Load Real Address	RS	B1	LRA	05.590
Load Rounded (Extended to Long)	RR	25	LRDR	01.415 - 00.230(NRD)
Load Rounded (Long to Short)	RR	35	LRER	01.185 - 00.345(NRD)
Load (Short)	RR	38	LER	00.380
Load (Short)	RX	78	LE	00.703
Monitor Call	SI	AF	MC	01.113+00.115(I2)+07.830(I) +02.576(EC)
Move (Immediate)	SI	92	MVI	00.530
Move (Characters) 16 Bytes or Less	SS	D2	MVC	Formula is dependent on length: 00.784+00.115(N)+00.702(N/4) +00.092(DW)
More than 16 Bytes				01.797+01.271(N/8)-00.805(N/16)
Move Long	RR	0E	MVCL	01.070+00.633(B/8) +00.547(N2/8) +04.439(B/256)
Move Numerics	SS	D1	MVN	00.406+00.115(N)+00.702(N/4) +00.593(N/8)
Move with Offset	SS	F1	MVO	00.923+00.196(N1)+00.782(N1/4) +00.403(C1)+00.115(N3)

Move Zones	SS	D3	MVZ	00.406+00.115(N)+00.702(N/4) +00.593(N/8)
Multiply	RR	1C	MR	01.645
Multiply	RX	5C	M	01.991
Multiply (Extended)	RR	26	MXR	19.0
Multiply Decimal	SS	FC	MP	-01.030+04.600(N1)-03.330(N1/4) +02.875(DW1)-01.150(N0)
Multiply Halfword	RX	4C	MH	01.416
Multiply (Long)	RR	2C	MDR	03.2
Multiply (Long)	RX	6C	MD	03.9
Multiply (Long to Extended)	RR	27	MXDR	08.7
Multiply (Long to Extended)	RX	67	MXD	09.9
Multiply (Short)	RR	3C	MER	01.8
Multiply (Short)	RX	7C	ME	02.1
OR	RR	16	OR	00.840
OR	RX	56	O	01.163
OR (Immediate)	SI	96	OI	00.829
OR (Characters)	SS	D6	OC	00.406+00.115(N)+00.702(N/4) +00.593(N/8)
Pack	SS	F2	PACK	01.585 + 00.230(N1/2) -00.012(DW1-C1)
Purge TLB	SI	B20D	PTLB	09.155
Read Direct	SI	85	RDD	Indeterminate
Reset Reference Bit	SI	B213	RRB	02.485
Set Clock	SI	B204	SCK	01.462
Set Clock Comparator	SI	B206	SCKC	04.498
Set CPU Timer	SI	B208	SPT	05.245
Set Prefix	S	B210	SPX	188.0
Set Program Mask	RR	04	SPM	00.610
Set PSW Key From Address	SI	B20A	SPKA	01.680

Set Storage Key	RR	08	SSK	01.530
Set System Mask	SI	80	SSM	01.416+00.805(EC)
Shift and Round Decimal	SS	F0	SRP	05.541+00.230(N)+01.018(N/4) +00.190(N/8) + 00.432(C1) -00.564(N8) + 00.104(N16) +00.863(DL) + 00.805(DR)
Shift Left Double	RS	8F	SLDA	02.945+00.230(P1)+00.805(S3)
Shift Left Double Logical	RS	8D	SLDL	01.105+00.230(S1)
Shift Left Single	RS	8B	SLA	01.795+00.115(S)-00.115(S0) +00.115(S5)+00.805(S3)
Shift Left Single Logical	RS	89	SLL	00.875-00.230(S)+00.115(S7) +00.115(S9)+00.115(S)(S9) +00.115(S)(S6)
Shift Right Double	RS	8E	SRDA	01.450+00.230(S1)+00.230(P1)
Shift Right Double Logical	RS	8C	SRDL	01.105+00.230(S1)
Shift Right Single	RS	8A	SRA	00.875+00.115(S1)
Shift Right Single Logical	RS	88	SRL	00.875-00.230(S) +00.115(S7) +00.115(S9)+00.115(S)(S9)
Signal Processor (See Note)	RS	AE	SIGP	Indeterminate
Start I/O	SI	9C00	SIO	Indeterminate
Start I/O Fast Release	SI	9C01	SIOF	Indeterminate
Store	RX	50	ST	00.645
Store Channel ID	SI	B203	STIDC	04.210
Store Character	RX	42	STC	00.760
Store Characters Under Mask	RS	BE	STCM	01.680
Store CPU Address	S	B212	STAP	02.370
Store CPU ID	SI	B202	STIDP	01.565
Store CPU Timer	SI	B209	STPT	02.025
Store Clock	SI	B205	STCK	02.428
Store Clock Comparator	SI	B207	STCKC	01.220
Store Control	RS	B6	STCTL	02.140+00.403(R) - 00.058(R1) +00.575(R/4)

Store Halfword	RX	40	STH	00.875
Store (Long)	RX	60	STD	00.875
Store Multiple	RS	90	STM	02.370+00.115(R)+00.920(QWB)
Store Prefix	S	B211	STPX	02.485
Store (Short)	RX	70	STE	00.875
Store Then AND System Mask	SI	AC	STNSM	02.255 + 00.920(EC)
Store Then OR System Mask	SI	AD	STOSM	02.255 + 00.920(EC)
Subtract	RR	1B	SR	00.495
Subtract	RX	5B	S	00.933
Subtract Decimal	SS	FB	SP	02.315+00.115(N1)+00.430(N1/4) +00.430(N2/4)+00.660(C1)
Subtract Halfword	RX	4B	SH	01.163+00.230(P2)
Subtract Logical	RR	1F	SLR	00.380
Subtract Logical	RX	5F	SL	00.703
Subtract Normalized (Extended)	RR	37	SXR	05.0
Subtract Normalized (Long)	RR	2B	SDR	02.2
Subtract Normalized (Long)	RX	6B	SD	02.5
Subtract Normalized (Short)	RR	3B	SER	02.0
Subtract Normalized (Short)	RX	7B	SE	02.4
Subtract Unnormalized (Long)	RR	2F	SWR	02.0
Subtract Unnormalized (Long)	RX	6F	SW	02.2
Subtract Unnormalized (Short)	RR	3F	SUR	01.8
Subtract Unnormalized (Short)	RX	7F	SU	02.1
Supervisor Call	RR	0A	SVC	05.107 + 02.415(EC)
Test and Set	SI	93	TS	01.565
Test Channel	SI	9F	TCH	Indeterminate
Test I/O	SI	9D	TIO	Indeterminate
Test Under Mask	SI	91	TM	00.703
Translate	SS	DC	TR	02.980+00.662(N)+01.725(DW)

Translate and Test	SS	DD	TRT	01.200 + 01.110(B)
Unpack	SS	F3	UNPK	01.613+00.202(N2/4)-00.028(C1)
Write Direct	SI	84	WRD	Indeterminate
Zero and Add	SS	F8	ZAP	02.315+00.115(N1)+00.430(N1/4) +00.430(N2/4)+00.660(C1)

Note: For the signal processor (SIGP) instruction, the condition code (CC) is set to 3 when:

1. The addressed CPU is not configured to the sending CPU (as when the system is in UP mode).
2. The addressed CPU has an address other than 0000 or 0001.
3. A malfunction occurs that prohibits the addressed CPU from responding.
4. Certain CE functions (such as single cycle or ROS stop) prohibit the addressed CPU from responding.

In items 1 and 2, the response time to set the condition code to 3 is in the microsecond range. In items 3 and 4, the response time to set the condition code to 3 is approximately 35 seconds.

The response time for most SIGP instructions is in the microsecond range, with the following exceptions:

Restart: While the response to the SIGP instruction is in the microsecond range, the CPU appears busy (CC = 2) to later SIGP instructions until the SVP goes to the program frame. This action may take several seconds.

Resets: The response to the SIGP instruction depends on the SVP frame. If the SVP is in the manual frame, the response is in the millisecond range. If the SVP is in the program frame, the response time is approximately 1.5 seconds.

TEST LIGHT

The Model 158 does not turn on the test light when the CE key is turned on.

INTEGRATED STORAGE CONTROLS (CPU FEATURE NO. 4650)

The control unit may exceed the 32-microsecond initial-selection time limit.

SYSTEM STATUS RECORDING

The Model 158 generates a system recovery machine-check interruption condition for certain situations that are not the result of machine malfunctions. Among these situations are the IPL procedure and the successful setting of the TOD clock by the instruction set clock. The interruption request due to setting the TOD clock is set at the time that the first carryout of the low-order portion of the clock occurs after the clock is set. This delay provides sufficient time for the control program to perform the time-dependent portion of TOD-clock synchronization in a multiprocessing system before the machine-check interruption occurs.

EMULATOR PER SUCCESSFUL BRANCH EVENT

The DIL instructions for the 14XX and 7074 emulator features on the 158-II do not cause the PER successful branch event to be indicated, even when the control is not returned to the next sequential instruction.

Appendix D. Glossary and Abbreviations

If the term you are seeking does not appear in this glossary, refer to *IBM Data Processing Glossary*, GC20-1699.

Address: An identification of a storage location or an I/O device.

Address Compare: A technique to stop the CPU at a specific address.

Address Modification: The process of changing the address part of a machine instruction via coded instructions.

Address Translation: The process of changing the address of an item of data or an instruction from its virtual address to its real storage address.

Alphameric: Pertaining to a character set that contains letters, digits, and special characters.

Attention Identifier (AID): A character that is set by the console in display mode when the operator presses ENTER or CANCEL, or detects a field with the light pen.

Attribute Character: A character that describes the characteristics of the data field that follows it.

Basic Control (BC) Mode: A mode in which the features of a System/360 computing system and additional System/370 features, such as certain new machine instructions, are operational on a System/370 computing system.

CCW: Channel command word.

CIDA: Channel indirect data addressing.

Control Registers: A set of registers used for operating system control of relocation, priority interruption, program event recording, error recovery, and masking operations.

CPU: Central processing unit.

CRT: Cathode-ray tube.

CSW: Channel status word.

CTCA: Channel-to-channel adapter.

Cursor Symbol: A short line (underscore) displayed on the CRT to indicate where the next character entered will be positioned.

DIDOCS: Device independent display operator console support.

DOS/VS: Disk operating system/virtual storage.

Dynamic Address Translation (DAT): (1) The translation of a virtual storage address to a real storage address during execution of an instruction. (2) A hardware feature that performs the translation.

EBCDIC: Extended binary-coded decimal interchange code.

EPO: Emergency power off.

Extended Control (EC) Mode: A mode in which all the features of a System/370 computing system, including dynamic address translation, are operational.

Hardstop: Faulty machine condition in which CPU ceases operation.

Hex: Denoted in the hexadecimal (base 16) number system.

HMS: Hierarchical monitoring system.

IAR: Instruction address register.

IMPL: Initial microprogram load.

Initialize: To set counters, switches, address, etc., to zero or other starting values at the beginning of, or at prescribed points in, a computer program.

IPL: Initial program load.

ISC: Integrated storage controls.

k: 1,000.

K: 1,024 bytes of storage capacity.

LCL: Limited channel logout.

Loosely Coupled: Pertaining to processors that are coupled by channel-to-channel adapters that are used to pass control information between the processors.

MCS: Multiple console support.

MP: Multiprocessor *or* the multiprocessor mode of operation.

Multiprocessor: A computer system having two processing units under a single control program.

MVS: Multiple virtual storage.

Offline: Pertaining to resources with which the central processing unit has no direct communication or control.

OLT(S)EP: Online test (standalone) executive program.

Online: Pertaining to resources with which the central processing unit has direct communication or control.

OS/VS: Operating system/virtual storage.

Page: (1) A fixed-length block of instructions, data, or both that can be transferred between real storage and external page storage. (2) To transfer instructions, data, or both between real storage and external page storage.

Page Table: A table that indicates whether a page is in real storage, and correlates virtual addresses with real storage addresses.

PER: Program event recording.

PFK: Program function key.

PSW: Program status word.

PTLB: Purge translation lookaside buffer.

RAS: Reliability and serviceability.

RCS: Reloadable control storage.

Real Address: The address of a location in real storage.

SAR: Storage address register.

SCU: Storage control unit.

Segment: A continuous 64K area of virtual storage, which is allocated to a job or system task.

Segment Table: A table used in dynamic address translation to control user access to virtual storage segments. Each entry indicates the length, location, and availability of a corresponding page table.

Selectable Unit: A collection of macro instructions and modules that provides added program function or hardware support for MVS.

SF: Start field.

Softstop: Stop condition in which CPU clock continues to run.

SVP: Service processor.

Tightly Coupled: Pertaining to processing units coupled by shared main storage.

TOD: Time of day.

UCW: Unit control word.

Uniprocessor: A computer system having a single processing unit.

UP: In an MP system, the uniprocessor mode of operation.

UPS: Uninterruptible power system.

Virtual Address: An address that refers to virtual storage and must, therefore, be translated into a real storage address when it is used.

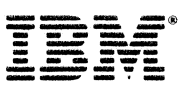
Wait State: The state of the system when no instructions are being processed, but the system is not in the stopped state. The system can accept I/O and external interruptions, and can be put through the IPL procedure.

Word: Predetermined group of bytes whose address is the first byte-address, located in storage by boundary limits; halfword = two bytes, address divisible by 2; fullword = four bytes, location divisible by 4; doubleword = eight bytes, location divisible by 8.

Write: To transfer data from main storage to an I/O device.

GA22-7011-5

IBM System/370 Model 158 Functional Characteristics (File No. S370-01) Printed in U.S.A. GA22-7011-5



International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601

IBM System/370 Model 158 Functional Characteristics

Order No. GA22-7011-5

READER'S COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name and mailing address:

_____

_____

_____

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure-sensitive or other gummed tape to seal this form.

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the front cover or title page.)

Reader's Comment Form

Cut or Fold Along Line

Fold and tape | Please Do Not Staple | Fold and tape

FIRST CLASS
PERMIT NO. 419
POUGHKEEPSIE,
NEW YORK

Business Reply Mail
No postage stamp necessary if mailed in U.S.A.

Postage will be paid by:

International Business Machines Corporation
Department B98
P.O. Box 390
Poughkeepsie, New York 12602

Fold and tape | Please Do Not Staple | Fold and tape

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601

IBM System/370 Model 158 Functional Characteristics (File No. S370-01) Printed in U.S.A. GA22-7011-5

IBM System/370 Model 158 Functional Characteristics

Order No. GA22-7011-5

READER'S COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name and mailing address:

_____

_____

_____

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure-sensitive or other gummed tape to seal this form.

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the front cover or title page.)

Reader's Comment Form

Cut or Fold Along Line

Fold and tape | Please Do Not Staple | Fold and tape

FIRST CLASS
PERMIT NO. 419
POUGHKEEPSIE,
NEW YORK

Business Reply Mail
No postage stamp necessary if mailed in U.S.A.

Postage will be paid by:

International Business Machines Corporation
Department B98
P.O. Box 390
Poughkeepsie, New York 12602

Fold and tape | Please Do Not Staple | Fold and tape

IBM System/370 Model 158 Functional Characteristics (File No. S370-01) Printed in U.S.A. GA22-7011-5

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601