File No. S360-01
Form A22-6884-1

IBM Systems Reference Library

IBM System/360 Model 65

Functional Characteristics

This manual presents the organization, characteristics, functions and features unique to the IBM System/360 Model 65. Major areas described are system structure, generalized information flow, standard and optional features, instruction timings, and the system control panel.

Descriptions of specific input/output devices used with the IBM System/360 Model 65 appear in separate publications. Configurations for the IBM 2065 Processing Unit and I/O devices are available. See *IBM System/360 Bibliography,* Form A22-6822.

It is assumed that the reader has a knowledge of the System/360 as defined in the *IBM System/360 Principles of Operation,* Form A22-6821 and the *IBM System/360 System Summary,* Form A22-6810.

This is a reprint of Publications Form No. A22-6884-0 incorporating the following Newsletter:

Form No.	Pages	Dated
N22-0211-1	5 thru 12	January 10, 1966

Copies of this and other IBM publications can be obtained through IBM Branch Offices.
Address comments concerning the contents of this publication to:
IBM Corporation, Product Publications, Dept. 528, Kingston, New York 12401

Contents

IBM System/360 Model 65

The IBM System/360 Model 65 is part of a series of distinguished, compatible, high performance, data processing systems. The Model 65 provides the reliability, convenience, and confidence, demanded by large scale business and scientific computation.

The Model 65 includes the advantages, characteristics and functional logic established for the System/360, as defined in the *IBM System/360 Principles of Operation,* Form A22-6821.

Major components comprising a System/360 Model 65 consist of an IBM 2065 Processing Unit, IBM 2365 Processor Storage, IBM 2361 Core Storage, IBM 2860 Selector Channels and IBM 2870 Multiplexor Channel with input/output devices attached to the channels through control units (Figure 1).

There are four models of the Model 65 termed G65, H65, I65 and J65. These four models differ only in the amount of 2365 Processor Storage required with a 2065 Processing Unit. The significant differences are:

IBM SYSTEM/360 MODEL	PROCESSING UNIT MODEL	DESCRIPTION
G65	2065G	Requires one 2365 Processor Storage Model 1 (131,072 bytes of storage)
H65	2065H	Requires one 2365 Processor Storage Model 2 (262,144 bytes of storage)
I65	2065I	Requires two 2365 Processor Storage Model 2 (524,288 bytes of storage)
J65	2065J	Requires four 2365 Processor Storage Model 2 (1,048,576 bytes of storage)

Outline configurations of the Model 65's, produced by the various combinations of a 2365 Processor Storage and a 2065 Processing Unit are shown in Figure 2.

The system control panel is located at one end of the 2065 Processing Unit. An optional IBM 1052 Printer-Keyboard may be placed adjacent to the 2065 Processing Unit reading board, to serve as an operator's console

The standard features for any System/360 Model 65 include:

Universal Instruction Set
Interval Timer
Storage Protection

Optional features for any System/360 Model 65 include:

2361 Core Storage (Model 1 and 2)
1052 Printer-Keyboard
Channel-to-Channel Feature
Direct Control Feature
2870 Multiplexor Channel
7070/7074 Compatability Feature (Models H65, I65, and J65 only)
7080 Compatability Feature (Models H65, I65, and J65 only)
709/7040/7044/7090/7094/7094II Compatability Feature (Models I65 and J65 only)

A variety of control units and input/output devices are available for use with the Model 65. Descriptions of specific input/output devices appear in separate publications. Configurators for the I/O devices and systems components are also available. See *IBM System/360 Bibliography,* Form A22-6822.

2365 Processor Storage

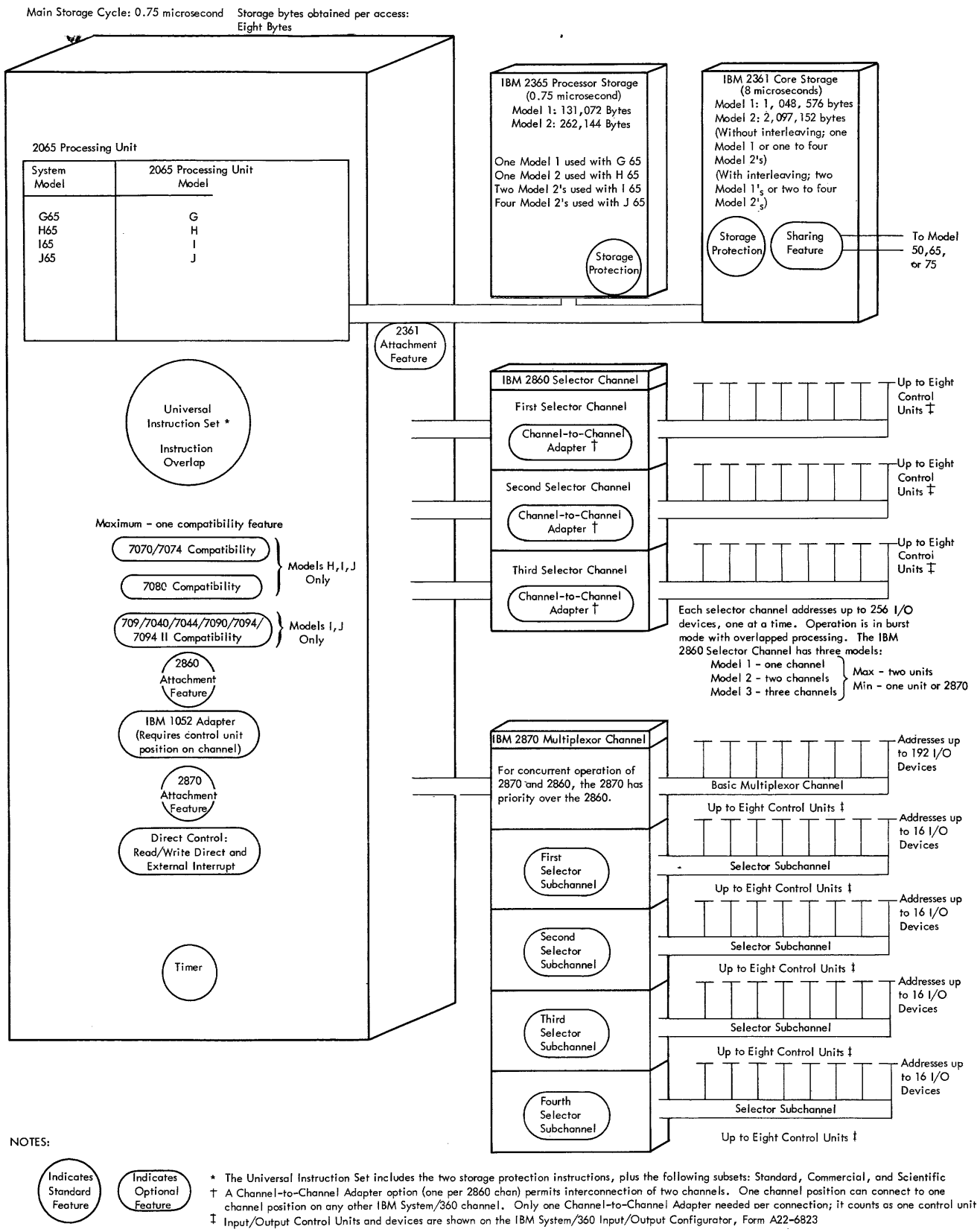
The 2365 Processor Storage Models 1 and 2, is the main storage for the Model 65. Model 1 has a storage capacity of 131,072 bytes and Model 2 has a storage capacity of 262,144 bytes.

All 2365 Processor Storage has a basic 750 nanosecond storage cycle, with access to eight bytes (double word) in parallel. A store function is possible on a byte basis and any number or combinations up to eight contiguous bytes can be stored in one storage cycle. Byte locations are consecutively numbered starting with zero. An addressing exception is recognized when any part of an operand is located beyond the maximum available main storage capacity.

Each 2365 Model 2 contains two independent storage units (arrays) each with its own address and storage buffer registers. Each array can access a double word (eight bytes) in parallel and has 131,072 bytes of storage organized into 16,284 double words. One array contains even numbered double words while the other array contains odd numbered double words. (Figure 3).

The two storage arrays are two-way interleaved. With interleaving the two storage arrays operate in an overlapped manner for improved sequential access. This means that two sequential (odd-even) storage double words (16 bytes) can be referred to (overlapped) at the same time. With interleaving, an effective sequential access rate of 400 nanoseconds per storage word (eight bytes) is possible.

In addition to the two-way interleaving (overlap) within a single 2365 Model 2, storage accesses to any two 2365's or 2361's can be overlapped. Each main storage unit, 2365 Processor Storage or 2361 Core Storage, used with the system operates independently of each other. The Storage units are related only in that they serve the common function of main storage and



NOTES:

Indicates Standard Feature

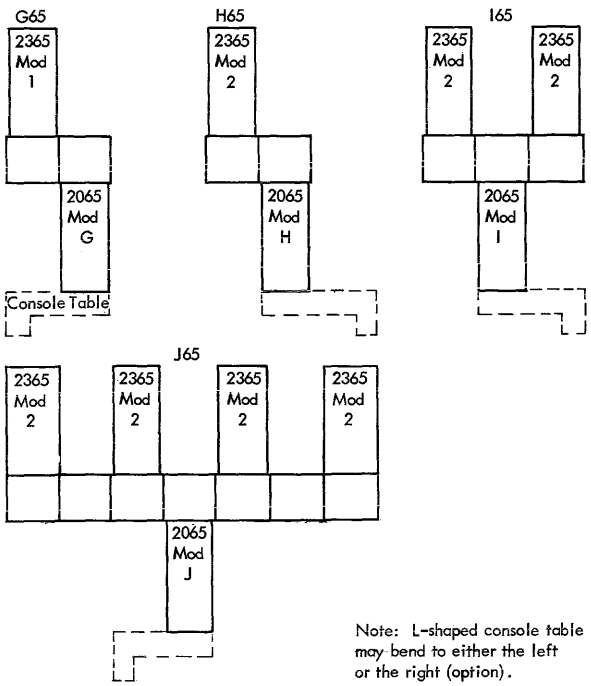
Indicates Optional Feature

* The Universal Instruction Set includes the two storage protection instructions, plus the following subsets: Standard, Commercial, and Scientific

† A Channel-to-Channel Adapter option (one per 2860 chan) permits interconnection of two channels. One channel position can connect to one channel position on any other IBM System/360 channel. Only one Channel-to-Channel Adapter needed per connection; it counts as one control unit

‡ Input/Output Control Units and devices are shown on the IBM System/360 Input/Output Configurator, Form A22-6823

Figure 1. System/360 Model 65 Configurator



● Figure 2. 2065 Processing Unit and 2365 Processor Storage Configurations

are assigned storage addresses that are contiguous from one unit to the next. Because each storage unit is independent, the access of information within the address range of each unit can be overlapped.

2361 Core Storage

The 2361 Core Storage is a large capacity direct access core storage unit. It has a basic 8 microsecond storage cycle, with access to two words (eight bytes) in parallel. The data access time (double word) is 3.6 microseconds. The remaining time is overlapped with execution and no further delay will occur unless the same storage unit is addressed during the remaining 4.4 microseconds. When the 2065 Processor addresses 2361 Storage, overlapped I/O references to 2365 Processor Storage are allowed. The reverse is also true.

The 2361 is an extension of the main storage (processor storage); addresses are contiguous with the 2365 Processor Storage addresses. The 2361 Model 1 has a storage capacity of 1,048,576 bytes and the Model 2 a storage capacity of 2,097,152 bytes.

A 2361 Core Storage can be shared with a System/360 Model 50, another Model 65 or a Model 75. When shared, 2361 addresses are an extension of the addresses of the larger of the two processor storages involved. Storage protection is a standard feature.

The 2361's can be specified for two-way interleaving. Interleaving provides an addressing scheme between

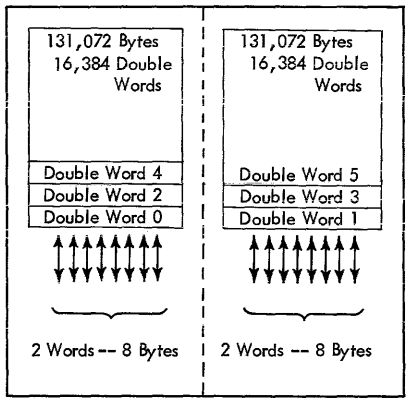


Figure 3. 2365 Processor Storage Model 2 Two-way Interleaving

two 2361's that permits the overlapping of read/write storage cycles in sequential operations. A sequential access rate of 4 microseconds per double word is possible and sequential access speeds of 2 megabytes (two million bytes) per second are possible.

One 2361 Model 1 or four 2361 Model 2's, without interleaving, can be used with Model 65. Two 2361 Model 1's or two or four 2361 Model 2's, with interleaving, can be used with Model 65. 2361's not equipped for two-way interleaving cannot be intermixed with 2361's equipped for interleaving.

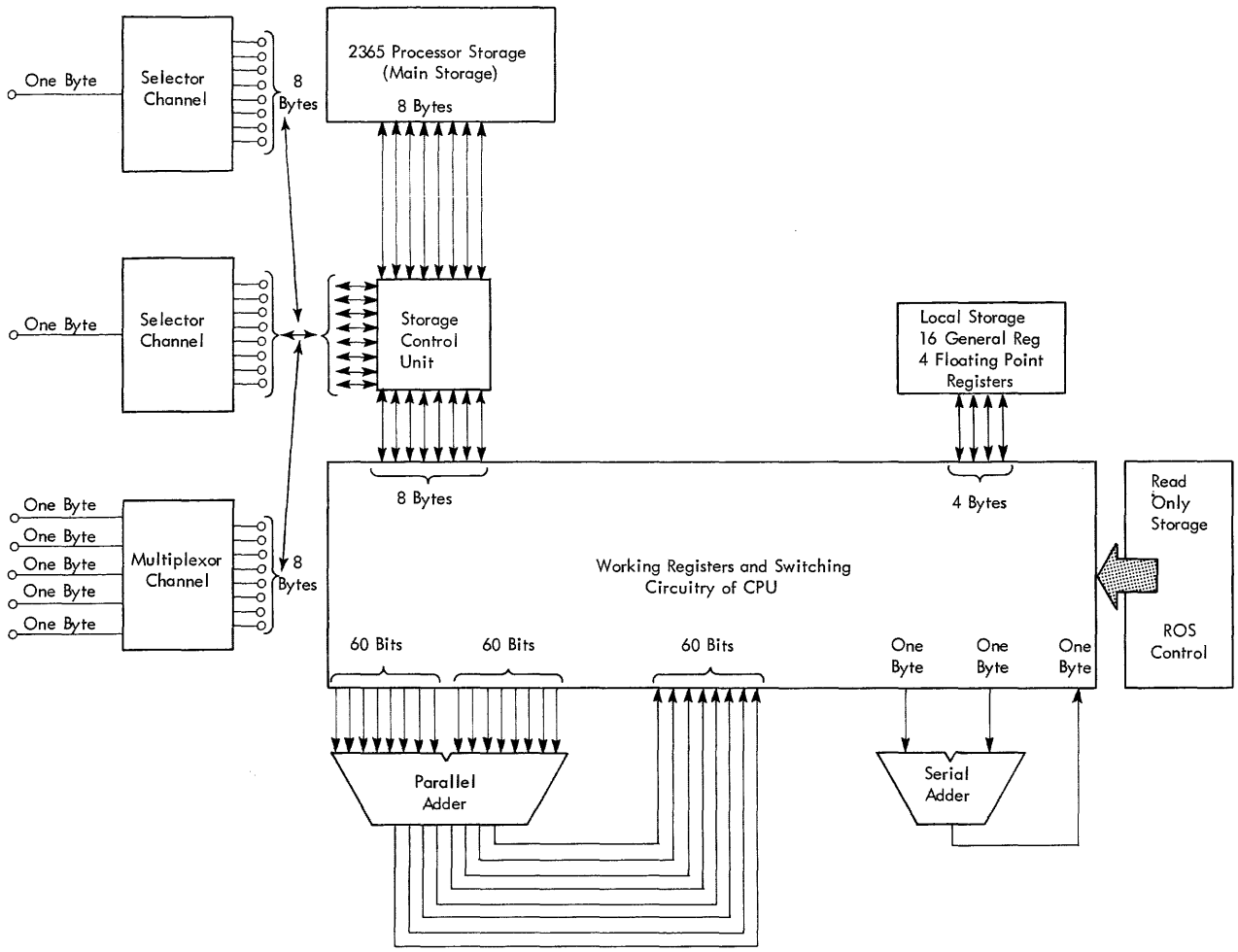
2065 Processing Unit

The 2065 Processing Unit is the central processing unit (CPU) for all Model 65's. The 2065 Processing Unit consists of data registers, interconnecting data paths and sequence controls. These facilities provide for addressing main storage (2365 Processor Storage), for fetching instructions in the desired order, and for initiating the communications between main storage and external devices (Figure 4).

The basic data path, with two arithmetic registers, allows for high speed and simplified implementation of the System/360 instruction set. In addition, an instruction buffer permits high-speed instruction preparation and overlap of most instruction fetch time.

The Model 65 is provided with the universal instruction set. The universal instruction set includes the standard instruction set, plus the instructions of the decimal feature, the floating-point feature, the storage protect feature, and the direct control feature.

Descriptions of all instructions are found in the *IBM System/360 Principles of Operation*. Timing informa-



	Data Width	Access/Speed/Rate	Comment
2365 Processor storage	8 bytes	.75 microsecond storage cycle	All models
2361 Core Storage	8 bytes	8 microsecond storage cycle	All models
General registers	1 word	200 nanoseconds	16 General registers
Floating-point registers	2 words	200 nanoseconds/word	4 Floating-point registers
Parallel adder	60 bits	200 nanoseconds	
Serial adder	1 byte	200 nanoseconds	
Basic machine cycle		200 nanoseconds	
2860 selector channel	1 byte	1.3 MC byte/sec	8 bytes to storage
2870 Multiplexor channel	1 byte	110 kc to 450 kc	8 bytes to storage
Burst mode	1 byte	110 kc	
Multiplexor mode	1 byte	110 kc	
Selector subchannel	1 byte	100 kc	

Figure 4. Model 65 Data Flow Diagram and System Statistics

tion for each of the instructions is found in the instruction timing section of this manual.

The 2065 Processing Unit contains the following major logical parts:

Arithmetic-logic units
Local store
  General registers
  Floating-point registers
Read only storage
Storage control unit

Arithmetic-Logic Unit

Two working arithmetic registers allow for high speed and simplified implementation of the arithmetic and logic operations. A parallel adder is a 60-bit wide binary adder used to facilitate handling of the long floating-point operations. The serial adder is an eight-bit decimal adder; also used as a binary adder. In addition, the logical functions OR, AND, and exclusive OR can be effected.

Local Store

Local store is a small high-speed storage unit providing registers for general machine working storage plus the general and floating-point registers. Only the general and floating-point registers are addressable by the main program.

General Registers

The general registers are used in address arithmetic and indexing, and as accumulators in fixed-point arithmetic and logical operations. The general-purpose registers have a capacity of one word (32 bits plus four parity bits). For some operations two adjacent registers can be coupled together providing a double word capacity. The general registers are implemented in local store and have a cycle time of .2 microsecond per four bytes.

Floating-Point Registers

Four floating-point registers are available for floating-point operations. These registers are two words (eight bytes) in length and can contain either a short (one word) or a long (two word) precision floating-point operand. The floating-point registers are implemented in local store and have a cycle time of .2 microsecond per four bytes.

Read Only Storage

The control function of the Model 65 is achieved by the use of a read only storage (ROS). The ROS contains predetermined information of a nondestructive nature, used to control the functions of data flow, instruction execution and is self addressable. ROS is not directly addressable by the main program. Modification of the unit is made by physically changing the ROS unit.

Storage Control Unit

The storage control unit handles all processor and channel references to main storage (2365 Processor Storage and 2361 Core Storage). The storage control unit operates in parallel with and is effectively independent of both the processor and all channels. Its function is to handle all requests for use of storage and to determine action to be taken in case of simultaneous requests either between channels or between channels and the processor (Figure 4). It is designed to maximize the number of storage references made by the channels or processor and to permit overlap of storage references, whenever possible.

Checking

Extensive checking capability is built into all units of the Model 65 based largely on a byte parity check. All data transfers are checked for correct parity both within and between units of the system. All storage references in either the 2365 Processor or 2361 Core Storage are checked for proper parity within the unit itself.

The 2065 Processing Unit includes checking of data transfers, arithmetic functions, as well as performing a parity check of the ROS control words.

The 2860 Selector Channel and 2870 Multiplexor Channel check parity of data transfers and in addition check the correctness of its arithmetic function.

Channels

The channel is the data path and direct controller of I/O control units and the I/O devices attached to the control units. The channel relieves the CPU of the task of communicating directly with the I/O devices and permits data processing to proceed concurrently with I/O operations.

Data are transferred a byte at a time between the I/O device and the channel. Data transfers between the channel and storage control unit are parallel by eight bytes for both selector and multiplexor channels (Figure 4).

A standard I/O interface provides a uniform method of attaching I/O control units to all channels, making the Model 65 adaptable to a broad spectrum of applications.

The 2860 Selector Channel and the 2870 Multiplexor Channel are available for the Model 65.

2860 Selector Channel

The 2860 Selector Channel provides for the attachment and control of a wide variety of burst mode I/O control

ciated devices. The 2860 is available in three models:

Model 1—provides one selector channel
Model 2—provides two selector channels
Model 3—provides three selector channels

Two 2860's in any combination of models can be attached to the processing unit. At least one 2860 (any model) or 2870 is required.

The selector channel permits data rates of 1.3 million bytes per second. I/O operations are overlapped with processing and depending on the data rate, all selector channels can operate concurrently. A full set of channel control and buffer registers permits each channel to operate with minimal interference.

A maximum of eight control units can be attached to each selector channel. Each channel may have more than one unit connected to it, but only one device per channel may transfer data at any given time.

Channel-to-Channel Feature

A channel-to-channel adapter is available as an optional feature. The adapter permits the communication between two System/360 channels, thus providing the capability for interconnection of two processing units within the System/360. The adapter uses one control unit position on each of the two channels. Only one of the two connected channels requires the feature. There can be a maximum of one channel-to-channel adapter per channel.

2870 Multiplexor Channel

The 2870 Multiplexor Channel provides for the attachment of a wide range of low to medium speed I/O control units and associated devices. One 2870 Multiplexor Channel can be attached to the Model 65.

The multiplexor channel provides up to 196 subchannels, including four selector subchannels. The basic multiplexor channel has 192 subchannels; it can attach eight control units and can address 192 I/O devices. The basic multiplexor channel can overlap the operation of several I/O devices in multiplex mode or operate a single device in burst mode. One to four selector subchannels are optional with a 2870. Each selector subchannel can operate one I/O device concurrently with the basic multiplexor channel. Each selector subchannel permits attachment of eight control units for devices having a data rate not exceeding 180 kb. Regardless of the number of control units attached, a maximum of 16 I/O devices can be attached to a selector subchannel.

The maximum aggregate data rate for the multiplexor channel ranges from 110 to 670 kb depending on the number of selector subchannels installed. Selector subchannels 1-3 may each operate concurrently at up to 180 kb; selector subchannel 4 has a maximum data-rate of 100 kb. Each selector subchannel in operation diminishes the basic multiplexor channel's maximum data-rate of 110 kb; the relationship to maximum data-rates for concurrent selector subchannel operations is shown in the following table:

BASIC MULTIPLEXOR CHANNEL	SELECTOR SUBCHANNELS 1ST	2ND	3RD	4TH
110kb				
88kb	180kb			
66kb	180kb	180kb		
44kb	180kb	180kb	180kb	
30kb	180kb	180kb	180kb	100kb

Note: The 180 kb maximum data-rate for selector subchannels pertains to attachment of magnetic tape devices; timing factors other than data-rates may preclude attachment of direct access storage devices having lesser data-rates.

Channel-to-Channel Adapter Connection to 2870

The 2870 may be connected to another system channel for channel-to-channel interconnection of two System/360 channels. The channel-to-channel adapter however, is installed on the other channel, not on the 2870.

2870 Priority

When both 2860 and 2870 channels are installed, the 2870 is connected to the end of the channel cable. When the 2870 operates concurrently with one or more 2860 channels, however, the 2870 has priority over the 2860 channels.

System Control Panel

The system control panel located at one end of the 2065 Processing Unit, provides the switches, the keys and the lights necessary to operate, monitor and control the Model 65. The need for operator manipulation of manual controls is held to a minimum by the system design and the governing supervisory program.

The operator control section of the system control panel can be duplicated once to provide a remote operator control panel that may be mounted on a 2150 Console or a 2250 Display Unit Model 1. An optional console typewriter input/output function can be provided by a 1052 Printer-Keyboard mounted either adjacent to the console table reading board or on a 2150 Console.

A detailed description of operator functions provided by the switches, keys and lights of the control panel is located in the system control panel section of this manual.

Interruption Times

Interruption times vary for the class of interruption and the type of instruction being executed at the time of the interruption.

External Interruption

External interruption time is 3.10 microseconds; it extends from the time the external interruption is discovered and honored to the next instruction.

Supervisor Call Interruption

Supervisor call interruption time is 3.75 microseconds; it extends from the time the external interruption is discovered and honored to the next instruction.

Program Interruption

Program interruption extends from the time the program interruption is discovered and honored to the next instruction. The program interruption time is equal to or less than the following time, plus the instruction time. The interruption time is 3.10 microseconds.

Machine Check Interruption

Machine check interruption time extends from the time the machine check interruption is discovered to the next instruction. The time is 50 microseconds and includes scan out and reset time.

I/O Interruption

The I/O interruption time extends from the time the CPU takes a pending interruption from the channel, to the storing of the old PSW and the CSW. This time is 5.4 microseconds.

System Control Panel

The system control panel contains the switches and lights necessary to operate, display, and control the system. The system consists of the CPU, storage, channels, on-line control units, and I/O devices. Off-line control units and I/O devices, although a part of the system environment, are not considered part of the system proper. (See Figure 5.)

System controls are logically divided into three classes: operator control, operator intervention, and customer engineering control. This section of the manual discusses the important system control functions provided by the system control panel as well as the purpose of the switches and lights on the panel.

By the use of the control panel, the operator can perform the following important system control functions:

1. Reset the system.
2. Store and display information in storage, registers, and program status word (PSW).
3. Load initial program information.

System Reset

The system reset function resets the CPU, channels, and on-line nonshared control units and I/O devices.

The CPU is placed in the stopped state and all pending interruptions are eliminated. All error-status indicators are reset to zero.

In general, the system is placed in such a state that processing can be initiated without the occurrence of machine checks, except those caused by subsequent machine malfunction.

The reset state for a control unit or device is described in the appropriate System Reference Library (SRL) publication. A system reset signal from a CPU resets only the functions in a shared control unit or device belonging to that CPU. Any function pertaining to another CPU remains undisturbed.

The system reset function is performed when the system reset key is pressed, when initial program loading is initiated, or when a power-on sequence is performed.

Programming Notes: If a system reset occurs in the middle of an operation, the contents of the PSW and of result registers or storage locations are unpredictable. If the CPU is in the wait state when the system reset is performed, and I/O is not operating, this uncertainty is eliminated.

A system reset does not correct parity in storage. Because a machine check occurs when information with incorrect parity is used, the incorrect information should be replaced by loading new information.

Store and Display

The store and display function permits manual intervention in the progress of a program. The storing and/or displaying of data may be provided by a supervisor program in conjunction with proper I/O equipment and the interrupt key.

In the absence of an appropriate supervisor program, the controls on the operator intervention panel allow storing and displaying of data directly. This is done by placing the CPU in the stopped state, and subsequently storing and/or displaying information in main storage, in general and floating-point registers, and in the instruction-address part of the PSW. The stopped state is achieved at the end of the current instruction when the stop key is pressed, when single instruction execution is specified, or when a preset address is reached. The store and display function is then achieved through the store, display, and set IC keys, address switches, data switches and storage select switch. Once the desired intervention is completed, the CPU can be started again.

All basic store and display functions can be simulated by a supervisor program. The stopping and starting of the CPU in itself does not cause any alteration in program execution other than the time element involved in the transition from operating to stopped state.

Machine checks occurring during store and display functions do not log immediately, but create a pending log condition that can be removed by a system reset or check reset. The error condition, when not disabled, forces a log-out and a subsequent machine check interruption when the CPU is returned to the operating state.

Initial Program Loading

Initial program loading (IPL) is provided for the initiation of processing when the contents of storage or the PSW are not suitable for further processing.

Initial program loading is initiated manually by selecting an input device with the load-unit switches and subsequently pressing the load key.

Pressing the load key causes a system reset, turns on the load light, turns off the manual light, and subsequently initiates a read operation from the selected input device. When reading is completed satisfactorily,

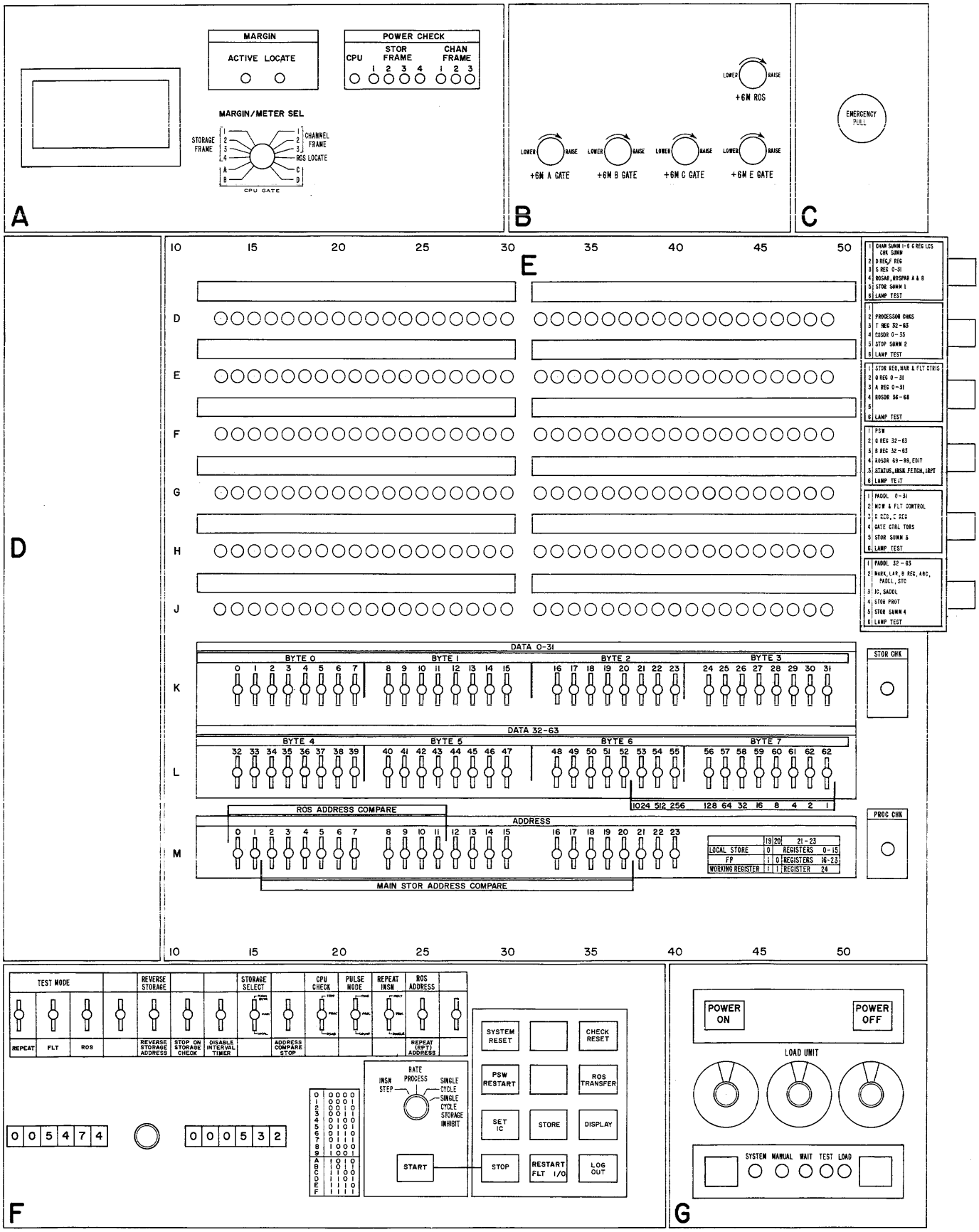


Figure 5. System Control Panel—Model 65

a new PSW is obtained, the CPU starts operating, and the load light is turned off.

The system reset suspends all instruction processing, interruptions, and timer updating and also resets all channels, on-line nonshared control units, and I/O devices. The contents of general and floating-point registers remain unchanged.

When IPL is initiated, the selected input device starts reading. The first 24 bytes read are placed in storage locations 0-23. Storage protection, program controlled interruption, and a possible incorrect length indication are ignored. The double word read into location 8 is used as the channel command word (CCW) for a subsequent I/O operation. When chaining is specified in this CCW, the operation proceeds with the CCW in location 16.

After the input operation is performed, the I/O address is stored in bits 21-31 of the first word in storage. Bits 16-20 are made zero. Bits 0-15 remain unchanged.

The CPU subsequently fetches the double word in location 0 as a new PSW and proceeds under control of the new PSW. The load light is turned off. When the I/O operations and PSW loading are not completed satisfactorily, the CPU idles, and the load light remains on.

Programming Notes: Initial program loading resembles a start I/O that specifies the I/O device selected in the load-unit switches and a zero protection key. The CCW for this start I/O is constructed in location 0, and contains a read command, zero data address, a byte count of 24, chain command flag on, suppress-length-indication flag on, program-controlled-interruption flag off, chain-data flag off and skip flag off.

Initial program loading reads new information into the first six words of storage. The remainder of the IPL program may be placed in any desired section of storage, but should not be placed in areas of storage reserved for the timer and PSW's.

If the selected input device is a disk, the IPL information is read from track 0.

The selected input device may be the channel-to-channel adapter involving two CPU's. A system reset on this adapter causes an attention signal to be sent to the addressed CPU. That CPU then should issue the write command necessary to load a program into main storage of the requesting CPU.

When the PSW in location 0 has bit 14 set to one, the CPU is in the wait state after the IPL procedure (the manual, the system and the load lights are off, and the wait light is on). Interruptions that become pending during IPL are taken before instruction execution.

System Control Panel Controls

System controls are divided into three logical groups identified as operator control, operation intervention and customer engineering control. Figure 5 shows the operator controls located in sections labeled C and G and operation intervention controls in sections F and E of the system control panel. The customer engineer will use all controls, but the controls in sections A, B, E, and F are intended primarily for customer engineering use.

Operator Controls

Sections C and G of the system control panel contain the controls required by the operator when the CPU is operating under full supervisor control. Under supervisor control, a minimum of direct manual intervention is required because the supervisor performs operations such as store and display.

The main functions provided by the operator controls are the control and indication of power, the indication of system status, operator to machine communication and initial program loading, the controls in section G are identical in all models of the System/360.

The following table lists all operator controls and indicator names and their implementation. All operator controls except the emergency pull switch are located in the section labeled G of the control panel shown in Figure 5. The emergency pull switch is located in section C.

NAME	IMPLEMENTATION
Emergency	Pull switch
Power On	Pushbutton switch (back lighted)
Power Off	Pushbutton switch
Interrupt	Pushbutton switch
Load	Pushbutton switch
Load Unit	Rotary switches (3)
Load	Indicator
Manual	Indicator
System	Indicator
Wait	Indicator
Test	Indicator

Emergency

Pulling this switch turns off all power beyond the entry terminal on every unit that is part of the system or that can be switched onto the system. Therefore, the switch controls the system proper, and all control units and I/O devices that are switched offline.

The switch latches in the out position and can be restored to its in position only by the customer engineer.

When the emergency pull switch is in the out position, the power on switch is ineffective.

Power On

This pushbutton switch initiates the power-on sequence for the system. At the completion of the power-on sequence, a system reset occurs. The button is backlighted to indicate when power is on. The switch is active only when the emergency pull switch is in its in position.

Power Off

This pushbutton switch initiates the power-off sequence for the system.

Interrupt

This pushbutton switch causes an external interruption request. The interruption is taken when not masked off and when the CPU is not stopped; otherwise, the interruption request remains pending. Bit 25 of the PSW is set to 1 when the interruption occurs, to indicate that the interrupt switch is the source of the external interruption.

Load

This pushbutton switch causes a system reset and starts the IPL procedure. For details of the IPL sequence, see *IBM System/360 Principles of Operation,* Form A22-6821.

Load Unit

These rotary switches (3) provide an 11-bit number to select the channel and I/O device to be used for IPL. The left switch has eight positions labeled 0 to 7 and selects the channel. The other two switches have 16 positions each, labeled with the standard hexadecimal characters 0-9 and A-F, and select the device.

Load

This light is on while the CPU is executing the initial program loading (IPL) function. The light is turned on when the load switch is pushed, and is turned off after the read operation and the loading of the new program status word (PSW) are completed successfully.

Manual

This light is on when the CPU is in the stopped state.

System

This light is on when the usage meter or customer engineering meter on the CPU cluster is running.

Wait

This light is on when the CPU is in the wait state. This is the case when bit 14 of the current PSW is 1.

Test

This light is on when a manual control is not in its normal position or when a diagnostic maintenance function is being performed for CPU, storage, or channels.

The following switches cause the test light to be on when not in their normal positions. The normal position for all rotary switches is straight up, and for all lever switches it is straight out.

Programming Note: The states indicated by the wait and manual lights are independent of each other; however, the state of the system light is not independent of the state of these two lights because of the definition of the running condition for the meters. The following table shows possible conditions:

SYSTEM LIGHT	MANUAL LIGHT	WAIT LIGHT	CPU STATE	I/O STATE
Off	Off	Off	Not allowed when power is on	
Off	Off	On	Waiting	Not operating
Off	On	Off	Stopped	Not operating
Off	On	On	Stopped, waiting	Not operating
On	Off	Off	Running	Undetermined
On	Off	On	Waiting	Operating
On	On	Off	Stopped	Operating
On	On	On	Stopped, waiting	Operating

Operator Intervention Controls

Sections F and E of the system control panel, contain the controls required for the operator to intervene in normal programmed system operation. These controls are intermixed with the customer engineering controls. Only operator intervention controls, on these panels, are described in this section.

Operator intervention controls provide the system reset and the store and display functions.

Start

This pushbutton provides a means of starting the machines in process, instruction step, single cycle, or single cycle storage inhibit, depending on the position of the rate switch. The operation is:

1. If start is depressed after a normal halt, instruction processing continues as if no halt had occurred. Interruptions pending will be taken after execution of the first instruction.

2. If start is depressed after an abnormal halt or system reset, the results will not necessarily be predictable.

3. The type of operation executed by the start pushbutton depends on the position of the rate switch described under the section pertaining to that switch.

Rate

This rotary switch has four positions labeled process, Insn step (instruction step), single cycle, and single cycle storage inhibit.

Process: The depression of the start pushbutton with the rate switch in the process position causes the machine to continue at normal speed as specified under the start pushbutton.

Insn Step: If set to the *Insn Step* position, the system executes one complete machine instruction for each depression of the start pushbutton. The operation is:

1. Any machine instruction can be executed in this mode. Interruptions are executed after the instruction is completed.

2. The stop point is identical to that achieved by the stop pushbutton.

3. When I/O operations are started, they will be completed to the interruption point.

4. The test light is on in this position.

Single Cycle: If set to *single cycle position* each depression of the start pushbutton advances the CPU by its minimum amount. The operation is:

1. When the instruction being single cycled utilizes asynchronous devices, it single cycles through all CPU functions of the instruction to the point of initiation of the asynchronous operation. The asynchronous operation starts on the next depression of the start pushbutton and runs to the completion point in a normal manner.

2. If the asynchronous device initiates an interruption request during single cycle operation it is not automatically executed. The interruption is broken into single operations. It requires more than one depression of the start pushbuttons to complete the transfer of PSW's.

3. The test light is on in this position.

Single Cycle Storage Inhibit: Customer engineering function.

Stop

This pushbutton provides the ability to completely terminate machine operations without destroying the machine environment. The operation is:

1. The CPU proceeds to the end of the machine instruction being executed at the time the stop command is recognized.

2. All waiting interruptions are executed.

3. Any I/O operation in process is allowed to be completed.

System Reset

This pushbutton resets the on-line channels, control units and CPU controls, including machine checks, to their initial state. The operation is:

1. Pushbutton is active in all modes of operation.
2. All check indicators are reset.
3. No data flow registers are reset.
4. The CPU is placed in the stopped state.

Storage Select (Section F)

This lever switch provides a means of selecting the storage unit that is to be addressed by the address keys when used in conjunction with the display or store pushbuttons. The operation is:

1. Normal position—main—selects the main storage for addressing when storing or displaying.

2. Down position—local store—selects the local store for addressing when storing or displaying.

3. Up position—main storage byte—selects the main storage for addressing when storing or displaying and causes only the byte addressed by the low three order address keys to be stored.

Address (Section E)

These 24 lever switches provide a manual means of selecting an addressable location in storage. The operation is:

1. Twenty-four keys are arranged in hexadecimal groups to permit storage addressing.

2. Correct parity is automatically generated.

3. Keys 2-20 are used in conjunction with the address compare switch for selection of an address for an address compare stop or address compare sync.

Data (Section E)

These 64 lever switches provide a manual means of entering data in the CPU. The operation is:

1. Sixty-four keys are arranged in hexadecimal groups to permit data entry.

2. Correct parity is automatically generated.

Store

This pushbutton provides a means of storing information in any address in the storage specified by the storage select toggle switch. The operation is:

1. The contents of the data keys are placed in the location specified by the address keys and the storage select switch.

2. Correct parity is automatically generated.

3. If the storage select switch is in the main position, the entire contents of the data keys are stored in the main storage.

4. If the storage select switch is in the local store position, the five low-order address keys specify the local store location in which the contents of the right half of the data keys will be stored. Address key 19 in the zero position permits storing in the general-purpose registers and in the 1 position permits storing in the floating-point registers. The specific address is determined by the five low-order address keys. Address keys 18 and 19 will address the working register when set to 1's.

5. If the storage select switch is in the main storage byte position, the byte of data in the data keys specified by the three low-order bits of the address keys will be stored in main storage at the address specified by the address keys.

6. The machine must be in a stopped state for this pushbutton to function.

Address Compare

This lever switch provides a machine stop on a CPU storage compare. The operation is:

1. In the center or normal position, a synchronizing pulse (for CE use) is provided whenever there is a com-

pare between the storage address bus and bits 2-20 of the address keys.

2. Down position—stop—the machine stops at the end of the instruction in progress, whenever there is a compare between the storage address bus and bits 2-20 of the address keys.

3. The test light is on whenever this switch is not in its normal position.

CPU Check

This lever switch provides a means of controlling the system when a machine check is encountered. The operation is:

1. Center position—process—upon detection of a machine check the CPU stops, the machine status is logged to storage and an interrupt trap is initiated if the machine check mask in the PSW is a 1. If the machine check mask in the PSW is a 0, the check is ignored except that the check triggers are turned on.

2. Up position—stop—upon the detection of a machine check, the check triggers are set. The CPU stops and no log-out occurs. If the check reset pushbutton is depressed, the operation is resumed but the results are not predictable.

Display

This pushbutton displays data specified by the combination of the storage select switch and the address keys. The operation is:

1. If the storage select switch is in the main position, the data in the main storage at the address in the address keys are displayed in the ST and AB registers. (See "Roller Indicators.")

2. If the storage select switch is in the local store position, the data in the local store are displayed in the T register. (See "Roller Indicators.")

3. The machine must be in a stopped state for this pushbutton to function.

Set IC (Instruction Counter)

This pushbutton provides a means of entering an address into the active program status word. The operation is:

1. The pushbutton sets the instruction counter portion of the PSW to the value specified in the address keys. The CPU is reset to the start of an I-fetch at the address. The instruction at the specified location is fetched and the instruction counter is updated.

2. The machine must be in a stopped state for this pushbutton to function.

NOTE: The instruction address is displayed in the D register when the CPU is in the stopped state. (See "Roller Indicators.") The new address contained in the instruction counter is one or two double words more than the instruction address contained in the address keys.

Restart PSW

This pushbutton switch provides a method to restart programs by loading a new program status word from the contents of storage location 0. The operation is:

1. With the machine in the stopped or reset state, depression of this pushbutton causes a new PSW to be fetched from storage location 0.

2. The CPU continues to process after the new PSW is fetched when the rate switch is in the process position.

Check Reset

This pushbutton provides a means of resetting all check indicators in the CPU. The operation is:

1. Depressing the pushbutton sets all CPU check indicators to the no check state (it is a subset of the system reset pushbutton).

2. All CPU logic check indicators on the system control panel are cleared as a result of Item 1.

3. If the CPU is stopped due to a machine check, processing continues when the check indicators are reset.

Log-Out

This pushbutton provides a means of logging the machine status into storage. The operation is:

1. The depression of this pushbutton causes the machine status to be stored in fixed locations in main storage.

2. This pushbutton is inactive under a normal processing condition. Log-out is the process of storing the status of most of the CPU indicators and registers in main storage. The log-out area occupies 44 words or 176 bytes of main storage, starting at byte 128.

Roller Indicators

Section E contains six rows of 36 indicator lights. Above each row of lights is an opening behind which is a roller that can be positioned to identify the significance of the related indicator lights for various operations. Each roller is manually placed in one of six positions by a positioning knob at the right side of panel B (Figure 5). The significance of each roller position is identified by the printing on the face of the panel by the related positioning knob.

In the display main storage operation, the contents of the addressed main storage locations are displayed in the ST and AB registers. For this operation, the ST register is identified on roller 1 position 3 and roller 2 position 3. The AB register is identified on roller 3 position 3 and roller 4 position 3.

In the display local storage operation, the contents of the addressed register are displayed in the T register. The T register is identified on roller 2 position 3.

When the CPU is in the stopped state, the instruction address is displayed in the D register. The D register is identified on roller 1 position 2.

The current PSW is identified on roller 4 position 1.

Customer Engineering Control

The ROS (read only storage) transfer and restart FLT (fault location test) pushbuttons as well as all lever switches, except storage select, are principally for customer engineering use. All switches, lights and the meter in sections A and B are for customer engineering use only.

Key Switch and Meters

The usage meter and a customer engineering meter are installed in section F of the system control panel. A key switch controls the meter to be run when the machine is in process. When power is on and the key switch is in the customer operation position, the usage meter accumulates time. If the key switch is in the customer engineer position, the CE meter accumulates time.

The instruction time tables presented in this bulletin are divided into two groups:

Group 1 This group of instruction times provides the average time for all instructions used with the Model 65.

All symbols used in the table of average instruction times should be interpreted in accordance with the Legend for System/360 Timing (Average Times).

Group 2 This group of instruction times contains the detailed timing formulas for all variable field length (VFL) instructions used with the Model 65.

All symbols used in the VFL formulas should be interpreted in accordance with the Legend for System/360 Timing (Detail VFL Times).

Within each group, timings are provided for instruction execution when instructions and data are located in main processor storage. All times are given in microseconds. Complete information for each instruction is included in the publication *IBM System/360 Principles of Operation*, Form A22-6821. Standard System/360 Timing Legends have been provided, therefore, all Legends given may not apply to the Model 65.

Timing Considerations

The following conditions (unless otherwise noted) were used in the development of these instruction time tables.

1. The time required for indexing by a base register is included in the times given. For those instructions that may be double indexed (indicated by one or two asterisks in the instruction name column), an additional .2 microsecond (two asterisks) or .10 microsecond (Model G—one asterisk) .15 microsecond (Models H, I, J—one asterisk) must be added to the times given in the table.

2. In all arithmetic operations, positive and negative operands are equally probable.

3. Each bit location has equal probability of containing bit values 0 or 1, and each bit location is independent of other bit locations.

Decimal data may contain digit values 0-9 in each digit position with equal probability.

4. Instructions may start on even or odd halfwords with equal probability.

5. Interruptions are not reflected in these timings.

6. All timings provided include both decoding and execution times for the instructions.

Timing Assumptions

The following assumptions (unless otherwise noted) were used in the development of the instruction time tables.

1. For add decimal (AP) and subtract decimal (SP) instructions, the first operand (i.e., the destination field) is assumed to be equal to or greater than the length of the second operand (i.e., the source field).

2. In the edit and mark (EDMK) instruction, an address is stored once. That is, this instruction is used with a signal field, or a line with only one numeric field is employed rather than a complete print line.

3. In the translate and test (TRT) instruction, it is assumed that a nonzero byte from a translate and test table is found.

4. The instruction times for floating-point instructions depend on the number of hexadecimal digits that are preshifted and post-shifted, as well as the number of times recomplementations of the result occurs. The times given in the tables for floating-point instructions are a *weighted average* of these variables.

5. For the pack (PACK), unpack (UNPK) and move with offset (MVO) instructions, it is assumed that no over-flow field occurs.

6. In the Model J65, add 50 nanoseconds to instruction times (RX, RS, SI, and ss formats) given for each reference to storage elements S3 and S4 for cable delay (Figure 2).

INSTRUCTION	FORM	MNEM	G	H, I, and J
Add	RR	AR	.65	.65
Add*	RX	A	1.50	1.40
Add Decimal	SS	AP	$3.6 + .2M + .2N_1 + .1N_2 + T_1$ ($2.0 + .4N_1$) $+ 1.2T_2$	$3.4 + .2M + .2N_1 + .1N_2 + T_1$ ($2.0 + .4N_1$) $+ 1.2T_2$
Add Halfword*	RX	AH	1.90	1.80
Add Logical	RR	ALR	.65	.65
Add Logical*	RX	AL	1.50	1.40
Add Normalized Long	RR	ADR	1.72	1.72
Add Normalized Long*	RX	AD	2.55	2.45
Add Normalized Short	RR	AER	1.68	1.68
Add Normalized Short*	RX	AE	2.53	2.43
Add Unnormalized Long	RR	AWR	1.65	1.65
Add Unnormalized Long*	RX	AW	2.50	2.40
Add Unnormalized Short	RR	AUR	1.64	1.64
Add Unnormalized Short*	RX	AU	2.48	2.38
AND	RR	NR	1.25	1.25
AND*	RX	N	2.10	2.00
AND	SI	NI	1.96	1.73
AND	SS	NC	$3.0 + .5N + .2N_3$	$2.8 + .5N + .2N_3$
Branch and Link	RR	BALR	1.25	1.20
Branch and Link**	RX	BAL	1.25	1.20
Branch on Condition	RR	BCR	$.7 + .5F_1$	$.7 + .4F_1$
Branch on Condition**	RX	BC	$.8 + .4F_1$	$.8 + .3F_1$
Branch on Count	RR	BCTR	$1.08 + .17F_1$	$.98 + .17F_1$
Branch on Count**	RX	BCT	1.25	1.15
Branch on Index High	RS	BXH	$1.6 - .2F_1$	$1.6 - .2F_1$
Branch on Index Low or Equal	RS	BXLE	$1.6 - .2F_1$	$1.6 - .2F_1$
Compare	RR	CR	.65	.65

INSTRUCTION	FORM	MNEM	G	H, I, and J
Compare*	RX	C	1.50	1.40
Compare Decimal	SS	CP	$3.67 + .2M + .13N_1 + .1N_2$	$3.47 + .2M + .13N_1 + .1N_2$
Compare Halfword*	RX	CH	1.90	1.80
Compare Logical	RR	CLR	.65	.65
Compare Logical*	RX	CL	1.50	1.40
Compare Logical	SI	CLI	1.50	1.40
Compare Logical	SS	CLC	$3.1 + .4B$	$2.9 + .4B$
Compare Long	RR	CDR	1.26	1.26
Compare Long*	RX	CD	2.10	2.00
Compare Short	RR	CER	1.24	1.24
Compare Short*	RX	CE	2.08	1.98
Convert to Binary*	RX	CVB	$7.5 + .2G_3 + .2G_1$	$7.4 + .2G_3 + .2G_1$
Convert to Decimal*	RX	CVD	$8.38 + .4G_3$	$8.05 + .4G_3$
Divide	RR	DR	$8.45 + .15G_1$	$8.45 + .15G_1$
Divide*	RX	D	$8.80 + .15G_1$	$8.70 + .15G_1$
Divide Decimal	SS	DP	$6.15 + 4.6N_1 - 4.7N_2 + 2.2N_2 (N_1 - N_2) + .6T_6 + 1.2T_7Q$	$5.95 + 4.5N_1 - 4.7N_2 + 2.2N_2 (N_1 - N_2) + .6T_6 + 1.2T_7Q$
Divide Long	RR	DDR	13.35	13.35
Divide Long*	RX	DD	14.20	14.10
Divide Short	RR	DER	6.55	6.55
Divide Short*	RX	DE	7.40	7.30
Edit	SS	ED	$3.40 + .7N + .1N_2$	$3.20 + .63N + .1N_2$
Edit and Mark	SS	EDMK	$3.40 + .63N + .1N_2 + 1.2MK$	$3.20 + .63N + .1N_2 + 1.2MK$
Exclusive OR	RR	XR	1.25	1.25
Exclusive OR*	RX	X	2.10	2.00
Exclusive OR	SI	XI	1.96	1.73
Exclusive OR	SS	XC	$3.0 + .5N + .2N_3$	$2.8 + .5N + .2N_3$
Execute*	RX	EX	$E_5 = 1.55 + E$ $E_6 = 3.20 + E$	$E_5 = 1.45 + E$ $E_6 = 3.0 + E$

INSTRUCTION	FORM	MNEM	G	H, I, and J
Halt I/O	SI	HIO	$1.5 + U_1 + U_2$	$1.4 + U_1 + U_2$
Halve Long	RR	HDR	1.25	1.25
Halve Short	RR	HER	1.05	1.05
Insert Character*	RX	IC	1.50	1.40
Insert Storage Key	RR	ISK	3.65	3.65
Load	RR	LR	.65	.65
Load*	RX	L	1.30	1.20
Load Address*	RX	LA	.90	.90
Load and Test	RR	LTR	.65	.65
Load and Test Long	RR	LTDR	1.05	1.05
Load and Test Short	RR	LTER	.85	.85
Load Complement	RR	LCR	.65	.65
Load Complement Long	RR	LCDR	1.05	1.05
Load Complement Short	RR	LCER	.85	.85
Load Halfword*	RX	LH	1.50	1.40
Load Long	RR	LDR	1.23	1.05
Load Long*	RX	LD	1.50	1.40
Load Multiple	RS	LM	$A_1 = 1.50$ $A_2 = .9 + .4R$ $A_3 = 1.3 + .4R$ $A_4 = 1.1 + .4R$	$A_1 = 1.40$ $A_2 = .8 + .4R$ $A_3 = 1.2 + .4R$ $A_4 = 1.0 + .4R$
Load Negative	RR	LNR	.95	.95
Load Negative Long	RR	LNDR	1.05	1.05
Load Negative Short	RR	LNER	.85	.85
Load Positive	RR	LPR	.95	.95
Load Positive Long	RR	LPDR	1.05	1.05
Load Positive Short	RR	LPER	.85	.85
Load PSW	SI	LPSW	2.40	2.20
Load Short	RR	LER	.65	.65
Load Short*	RX	LE	1.30	1.20
Move	SI	MVI	1.56	1.33

INSTRUCTION	FORM	MNEM	G	H, I, and J
Move	SS	MVC	$V_1 = 2.2 + 1.6W$ $V_2 = 3.3 + .2N + .2N_6$ $V_3 = 3.13 + .38N + .2N_3$	$V_1 = 2.0 + 1.6W$ $V_2 = 3.1 + .2N + .2N_6$ $V_3 = 2.93 + .38N + .2N_3$
Move Numerics	SS	MVN	$3.0 + .5N + .2N_3$	$2.8 + .5N + .2N_3$
Move With Offset	SS	MVO	$3.13 + .27N_1 + .3N_2 + .4N_3$ (if $N_1 \geqq N_2$)	$2.93 + .27N_1 + .3N_2 + .4N_3$ (if $N_1 \geqq N_2$)
Move Zones	SS	MVZ	$3.0 + .5N + .2N_3$	$2.8 + .5N + .2N_3$
Multiply	RR	MR	4.45	4.45
Multiply*	RX	M	4.90	4.80
Multiply Decimal	SS	MP	$3.97 + 3.5N_1 - 2.9N_2 + 1.0N_2 (N_1 - N_2)$	$3.77 + 3.4N_1 - 2.9N_2 + 1.0N_2 (N_1 - N_2)$
Multiply Halfword*	RX	MH	5.10	5.00
Multiply Long	RR	MDR	7.25	7.25
Multiply Long*	RX	MD	7.70	7.60
Multiply Short	RR	MER	4.05	4.05
Multiply Short*	RX	ME	4.50	4.40
OR	RR	OR	1.25	1.25
OR*	RX	O	2.10	2.00
OR	SI	OI	1.96	1.73
OR	SS	OC	$3.0 + .5N + .2N_3$	$2.8 + .5N + .2N_3$
Pack	SS	PACK	$3.07 + .27N_1 + .2N_2$ $\left(\text{if } N_1 \geqq \frac{N_2 + 1}{2}\right)$	$2.87 + .27N_1 + .2N_2$ $\left(\text{if } N_1 \geqq \frac{N_2 + 1}{2}\right)$
Read Direct	SI	RDD	$2.55 + ED$	$2.45 + ED$
Set Program Mask	RR	SPM	.85	.85
Set Storage Key	RR	SSK	2.6	2.6
Set System Mask	SI	SSM	1.90	1.80
Shift Left Double	RS	SLDA	$.9 + .4Q_1 + .2S_3$	$.9 + .4Q_1 + .2S_3$
Shift Left Double Logical	RS	SLDL	$.9 + .4Q_1 + .2S_3$	$.9 + .4Q_1 + .2S_3$
Shift Left Single	RS	SLA	$.7 + .2Q_1 + .2S_1$	$.7 + .2Q_1 + .2S_1$
Shift Left Single Logical	RS	SLL	$.7 + .2Q_1 + .2S_1$	$.7 + .2Q_1 + .2S_1$

INSTRUCTION	FORM	MNEM	G	H, I, and J
Shift Right Double	RS	SRDA	$.9 + .4Q_1 + .2S_4$	$.9 + .4Q_1 + .2S_4$
Shift Right Double Logical	RS	SRDL	$.9 + .4Q_1 + .2S_4$	$.9 + .4Q_1 + .2S_4$
Shift Right Single	RS	SRA	$.9 + .2Q_1 + .2S_2$	$.9 + .2Q_1 + .2S_2$
Shift Right Single Logical	RS	SRL	$.9 + .2Q_1 + .2S_2$	$.9 + .2Q_1 + .2S_2$
Start I/O	SI	SIO	$1.50 + U_1$	$1.40 + U_1$
Store*	RX	ST	1.16	.93
Store Character*	RX	STC	1.56	1.33
Store Halfword*	RX	STH	1.96	1.73
Store Long*	RX	STD	1.16	.93
Store Multiple	RS	STM	$A_1 = 1.56$ $A_2 = .76 + .4R$ $A_3 = 1.56 + .4R$ $A_4 = 1.16 + .4R$	$A_1 = 1.33$ $A_2 = .53 + .4R$ $A_3 = 1.33 + .4R$ $A_4 = .93 + .4R$
Store Short*	RX	STE	1.16	.93
Subtract	RR	SR	.65	.65
Subtract*	RX	S	1.50	1.40
Subtract Decimal	SS	SP	$3.6 + .2M + .2N_1$ $+ .1N_2 + T_1$ $(2.0 + .4N_1)$ $+ 1.2T_2$	$3.4 + .2M + .2N_1$ $+ .1N_2 + T_1$ $(2.0 + .4N_1)$ $+ 1.2T_2$
Subtract Halfword*	RX	SH	1.90	1.80
Subtract Logical	RR	SLR	.65	.65
Subtract Logical*	RX	SL	1.50	1.40
Subtract Normalized Long	RR	SDR	1.72	1.72
Subtract Normalized Long*	RX	SD	2.55	2.45
Subtract Normalized Short	RR	SER	1.68	1.68
Subtract Normalized Short*	RX	SE	2.53	2.43
Subtract Unnormalized Long	RR	SWR	1.65	1.65
Subtract Unnormalized Long*	RX	SW	2.50	2.40

INSTRUCTION	FORM	MNEM	G	H, I, and J
Subtract Unnormalized Short	RR	SUR	1.64	1.64
Subtract Unnormalized Short*	RX	SU	2.48	2.38
Supervisor Call	RR	SVC	4.15	3.75
Test Channel	SI	TCH	$1.50 + U_1$	$1.40 + U_1$
Test I/O	SI	TIO	$1.50 + U_1$	$1.40 + U_1$
Test Under Mask	SI	TM	1.70	1.60
Translate	SS	TR	$2.14 + 1.88N$	$1.94 + 1.78N$
Translate and Test	SS	TRT	$4.3 + 1.3B - 1.0T_9$	$4.1 + 1.2B - 1.0T_9$
Unpack	SS	UNPK	$3.0 + .4N_1 + .1N_2 + .2T_3$ $\left(\text{if } \frac{N_1+1}{2} \geq N_2\right)$	$2.8 + .4N_1 + .1N_2 + .2T_3$ $\left(\text{if } \frac{N_1+1}{2} \geq N_2\right)$
Write Direct	SI	WRD	1.90	1.80
Zero and Add	SS	ZAP	$3.70 + .2M + .1N_1 + .1N_2 + .72T_8 + 1.2T_2$	$3.50 + .2M + .1N_1 + .1N_2 + .72T_8 + 1.2T_2$
Test and Set	SI	TS	1.90	1.80

Legend for System/360 Timing (Average Times)

Fixed and Floating-Point Arithmetic, Logical and Branching Operations

A_1: Use if the number of registers is 2, and if the operand lies on double word boundaries

A_2: Use if the number of registers is even, and if the operand lies on double word boundaries

A_3: Use if the number of registers is even, and if the operand does not lie on double word boundaries

A_4: Use if the number of registers is odd

$F_1 = 1$ if the branch is successful
$= 0$ otherwise

$F_2 = 0$ if the R_2 field is zero (i.e., branch is suppressed)
$= 1$ otherwise

$G_1 = 1$ if an overflow (or divide check) occurs
$= 0$ otherwise

$G_2 = 1$ if overflow occurs and interrupt is masked
$= 0$ otherwise

$G_3 = 0$ if operand is positive
$= 1$ otherwise

$G_4 = 1$ if all of the selected bits are zero
$= 0$ otherwise

H_2 = number of high-order hexadecimal zeros in the second operand

Define Q_i and R_i: $H_2 = 2Q_4 + R_4$

$Q_x = \text{Quotient}_x$

$R_x = \text{Remainder}_x$

K_1 = Number of zero hexadecimal digits in the absolute value (i.e., integral value) of the multiplier field. The field with the smallest absolute value is used as the multiplier. In halfword multiply, K_1 applies only to the 16 low-order bits of the field with the smallest absolute value.

R = number of registers loaded or stored

$T_4 = 1$ if the second operand has leading hexadecimal zeros
$= 0$ otherwise

$T_5 = 9.38$ if signs differ
$= 10.0$ if signs are alike, and the high-order 16 bits of the first operand are significant
$= 10.63$ if inequality is found in byte 2
$= 11.25$ if inequality is found in byte 3, or if comparison is equal

Execute Instruction

E_1: Use when executed instruction is one halfword long

E_2: Use when executed instruction is two halfwords long

E_3: Use when executed instruction is a three-halfword character instruction

E_4: Use when executed instruction is a three-halfword decimal instruction

A_5: Use if leading 16 bits are not changed by operation

A_6 Use if leading 16 bits are changed by operation

E_5: Use if subject instruction is a successful branch

E_6: Use if subject instruction is not a successful branch

E = time for the instruction to be executed by the execute instruction

$T_{12} = 1$ if $R_1 \neq 0$
$= 0$ otherwise

Convert Instructions

C_1: Use when the number converted contains eight or less decimal digits

C_2: Use when the number converted contains more than eight decimal digits, but less than seven hexadecimal digits

C_3: Use when the number converted contains more than seven hexadecimal digits

$G_1 = 1$ if an overflow (or divide check) occurs
$= 0$ otherwise

$G_3 = 0$ if operand is positive
$= 1$ otherwise

H = number of significant (i.e., other than high-order zeros) hexadecimal digits in the binary operand

Define Q_3 and R_3: $8 - H = 2Q_3 + R_3$

$H_1 = 28.13$ if there are four leading zero bytes
$= 40.0$ if there are three leading zero bytes
$= 53.75$ if there are two leading zero bytes
$= 72.50$ if there is one leading zero byte
$= 95.63$ if there are no leading zero bytes

$T_1 = 1$ if the result field is recomplemented (i.e., changes sign)
$= 0$ otherwise

Binary Shift Operations

Let S = amount to be shifted

S(x) is a variable function defined as:

$S(x) = 1$ if $x = 0$
0 if $x \neq 0$

Define Q_1, R_1, Q_2, and R_2:

$S = 4Q_1 + R_1 = 8Q_2 + R_2$; where $0 \leq R_1 \leq 3$; $0 \leq R_2 \leq 7$; that is, Q_1 R_1 and Q_2 R_2 is the quotient and remainder found by dividing S by 4 and 8 respectively.

Define cases of S:

$S_1 = 1$ if $R_1 = 3$, or if $Q_1 = 0$
$= 2$ if $R_1 = 3$ and $Q_1 = 0$
$= 0$ otherwise

$S_2 = -1$ if $R_1 = 0$
$= 1$ if $R_1 = 1$, and $Q_1 = 0$
$= 0$ otherwise

$S_3 = 0$ if $R_1 = 0$, and $Q_1 \neq 0$
$= 1$ if $R_1 = 0$, and $Q_1 = 0$
$= 3$ if $R_1 = 1$
$= 5$ if $R_1 = 2$ or 3

$S_4 = 0$ if $R_1 = 0$
$= 4$ if $Q_1 = 0$ and $R_1 = 1$, or if $Q_1 \neq 0$ and $R_1 = 2$
$= 3$ if $Q_1 = 0$ and $R_1 = 2$, or if $Q_1 \neq 0$ and $R_1 = 3$
$= 2$ if $Q_1 = 0$ and $R_1 = 3$
$= 5$ if $Q_1 \neq 0$ and $R_1 = 1$

$S_5 = 1$ if high-order part of the result is zero
$= 0$ otherwise

$S_6 = 1$ if operand is negative
$= 0$ otherwise

$S_7 = 1$ if $R_1 \neq 0$ and operand is negative
$= 0$ otherwise

$S_8 = 1$ if $R_2 = 3, 4, 5$, or 6
$= 2$ if $R_2 = 7$
$= 0$ if $R_2 = 0, 1$, or 2

$S_9 = 1$ if $R_2 = 2, 3, 4$, or 5
$= 2$ if $R_2 = 1$
$= 0$ if $R_2 = 0, 6$, or 7

$S_{10} = 0$ if $0 \leq S \leq 7$
$= 1$ if $8 \leq S \leq 15$
$= 2$ if $16 \leq S \leq 23$
$= 3$ if $24 \leq S \leq 31$
$= 4$ if $32 \leq S \leq 39$

= 5 if $40 \leq S \leq 47$
= 6 if $48 \leq S \leq 55$
= 7 if $56 \leq S \leq 63$
where the symbol "$\leq$" means "less than or equal to"

Variable Field Length Instructions—Average Times

B = total number of bytes of the first operand which are processed (applies to instructions with a single length)

M = maximum of N_1 and N_2

MK = number of times the mark address is stored in the edit and mark instruction

N = total number of bytes in the first operand and for those instructions with a single length

N_1 = total number of bytes in the first operand

N_2 = total number of bytes in the second operand

N_3 = total number of bytes which overlap between the first and second operands
= 0 for nonoverlapping fields, and for overlapping fields where the address of the second operand is greater than or equal to ($\geq$) the first operand address

N_4 = total number of field separator characters in the edit pattern

N_5 = total number of control characters in the edit pattern

N_6 = number of bytes of the field which lie outside of that part of the field bounded by double words

N_7 = greatest integer of $\dfrac{(N_1 - 1)}{8}$

Q = minimum of $N_1 - 8$ and $N_1 - N_2$

SG = number of signs in field(s) to be edited

$T_1 = 1$ if the result field is recomplemented (i.e., changes sign)
= 0 otherwise

$T_2 = 1$ if the result field is zero
= 0 otherwise

$T_3 = 1$ if $N_2 < \frac{1}{2}(N_1 + 1)$
= 0 otherwise

$T_6 = 0$ if $N_2 \leq 4$
= 1 otherwise

$T_7 = 0$ if $N_1 \leq 8$
= 1 otherwise

$T_8 = 0$ if fields do not overlap
= 1 otherwise

$T_9 = 0$ if any nonzero function byte is found
= 1 otherwise

$T_{15} = 1$ if B = N and operands are equal
= 0 otherwise

$T_{16} = 0$ if $N_1 \geq N_2$
= 1 otherwise

$T_{17} = 1$ if $N_1 > \dfrac{N_2}{2}$
= 0 otherwise

$T_{18} = 1$ if N = 1
= 0 otherwise

$T_{19} = 1$ if $N_1 > 2N_2$
= 0 otherwise

V = absolute value (i.e., unsigned value) of $N_1 - N_2$

V_1: Use if first and second operand fields start and end on double word boundaries

V_2: Use if first and second operand fields start at corresponding byte addresses within double words but do not lie on double word boundaries

V_3: Use if first and second operand fields do not start at corresponding byte addresses within double words or if $N < 8$

V_4: Use if first and second operands fields start on double word boundaries but do not end on double word boundaries. N must be greater than seven to use this case.

NOTE: A byte address with a double word can have the value 0, 1, 2, 3, 4, 5, 6, or 7.

W = total number of double words in the first operand for those instructions with a single length

Input/Output Operations

B_1: Use when addressing the multiplexor channel in the multiplex mode

B_2: Use when addressing the multiplexor channel in the burst mode

B_3: Use when addressing the selector channel

D_1: Use if the multiplexor channel is busy and in the multiplex mode

D_2: Use if the multiplexor channel is busy and in the burst mode

D_3: Use if the multiplexor channel is idle

D_4: Use if the multiplexor channel has an interrupt pending

D_5: Use if the selector channel is busy

D_6: Use if the selector channel is idle

D_7: Use if the selector channel has an interrupt pending

ED = external delay

U_1 = select out delay plus device delay

U_2 = device delay for halt I/O sequence

Models H, I and J Variable Field Length Instructions—Timing Formulas—Detailed Times

In the following timing formulas, the times for the variable field length instructions (i.e., those instructions that contain an "L" field) are given in terms of word boundary crossovers and the operand addresses. The term "word boundary" is used to specify the boundary between two physical words. A physical word is the amount of information fetched in a single storage cycle (this is 64 bits for all Model 65's). Thus, the number of word boundary crossovers is one less than the number of double words spanned by the field.

All symbols used in the following VFL detailed timing formulas should be interpreted in accordance with the Legend for System/360 Timing (Detail VFL Times).

ADD Decimal—AP

$$AP = 3.75 + .2M + 1.6NWBL_1 + .8NWBL_2 + T_1 (2.2 + 2N_1 + 1.6NWBL_1) + 1.2T_2$$

AND—NC

$$NC = 3.16 + .2N + 1.6NWBL_1 + .8NWBL_2 + .2N_3$$

Compare Decimal—CP

$$CP = 3.75 + .2M + 1.00NWBL_1 + .8NWBL_2$$

Compare Logical—CLC

$$CLC = 3.16 + .2B + .8NWBB_1 + .8NWBB_2$$

NOTE: The compare logical operation is terminated when an unequal condition is found.

Divide Decimal—DP

$$DP = 6.2 + 4.4N_1 - 4.8N_2 + 2.20N_2 (N_1 - N_2) + .8NWBL_1 + .8NWBL_2 + .6T_6 + 1.2T_7Q$$

EDIT—ED

$$ED = 3.57 + .43N + 1.6NWBL_1 + .8NWBL_2$$

EDIT and MARK—EDMK

$$EDMK = 3.56 + .43N + 1.6NWBL_1 + 1.2NWBL_2 + 1.2MK$$

Exclusive OR—XC

$$XC = 3.16 + .2N + 1.6NWBL_1 + .8NWBL_2 + .2N_3$$

Move Characters—MVC

V_1: $2.0 + 1.6W$

V_2: $3.36 + .8NWBL_1 + .8NWBL_2 + .2N_6$

V_3: $3.16 + .6NWBL_1 + .8NWBL_2 + .2N_3 + .2N$

Move Numerics—MVN

$$MVN = 3.16 + .2N + 1.6NWBL_1 + .8NWBL_2 + .2N_3$$

Move with Offset—MVO

$$MVO = 3.16 + .2N_1 + .2N_2 + .6NWBL_1 + .8NWBL_2 + .4N_3$$

Move Zones—MVZ

$$MVZ = 3.16 + .2N + 1.6NWBL_1 + .8NWBL_2 + .2N_3$$

Multiply Decimal—MP

$$MP = 4.12 + 3.4N_1 - 3.2N_2 + 1.0N_2 (N_1 - N_2) + .8NWBL_2 + 1.6NWBL_1L_2$$

OR—OC

$$OC = 3.16 + .2N + 1.6NWBL_1 + .8NWBL_2 + .2N_3$$

Pack—PACK

$$PACK = 3.11 + .2N_1 + .1N_2 + .6NWBL_1 + .8NWBL_2$$

Subtract Decimal—SP

$$SP = 3.75 + .2M + 1.6NWBL_1 + .8NWBL_2 + T_1 (2.2 + .2N_1 + 1.6NWBL_1) + 1.2T_2$$

Translate—TR

$$TR = 2.15 + 1.6N + 1.4NWBL_1$$

Translate and Test—TRT

$$TRT = 5.35 + 1.08 + 1.6NWBB_1 - 1.00T_9$$

Unpack—UNPK

$$UNPK = 3.16 + .2N_1 + 1.6NWBL_1 + 1.6NWBL_2 + .2T_3$$

Zero and Add—ZAP

$$ZAP = 3.75 + .2M + .8NWBL_1 + .8NWBL_2 + .72T_8 + 1.2T_2$$

Legend for System/360 Timing (Detail VFL Times)

ABV = absolute value (i.e., integer value) of $NWBL_1 - NWBL_2$

B = total number of bytes in the first operand, which are processed (applies to instructions with a single length)

DBA = destination byte address within a word. (Note: A byte address within a word is 0, 1, 2, or 3)

G_2 = 1 if an overflow occurs and the interrupt is masked
= 0 otherwise

HB_1 = 1 if the address of the high-order (left-most) byte of the first operand is odd
= 0 otherwise

HB_2 = 1 if the address of the high-order (left-most) byte of the second operand is odd
= 0 otherwise

LB_1 = 1 if the address of the low-order (right-most) byte of the first operand is odd
= 0 otherwise

LB_2 = 1 if the address of the low-order (right-most) byte of the second operand is odd
= 0 otherwise

M = maximum of N_1 and N_2

MK = number of times the mark address is stored in the edit and mark instruction

MQ_1 = 0 is multiplier or quotient lies on a word boundary
= 1 otherwise

N = total number of bytes in the first operand, and for those instructions with a single length

N_1 = total number of bytes in the first operand

N_2 = total number of bytes in the second operand

N_3 = total number of bytes which overlap between the first and second operands
= 0 for nonoverlapping fields, and for overlapping fields where the address of the second operand is $\geq$ first operand address

N_4 = total number of field separator characters in the edit pattern

N_5 = total number of control characters in the edit pattern

N_6 = number of bytes of the field which lie outside of that part of the field bounded by double words

$NWBB_1$ = number of word boundary crossovers for that part of the first operand processed

$NWBB_2$ = number of word boundary crossovers for that part of the second operand processed

$NWBL_1$ = number of word boundary crossovers for the first operand

$NWBL_1L_2$ = number of word boundary crossovers for that part of the first operand which consists of N_2 bytes of high-order zeros

$NWBL_2$ = number of word boundary crossovers for the second operand

$NWBQ_1$ = number of word boundary crossovers for the quotient field

$NWBR_1$ = number of word boundary crossovers for the remainder field

Q = minimum of $N_1 - 8$ and $N_1 - N_2$

SBA = source byte address within a word

SG = number of signs in the field to be edited

T_1 = 1 if the result field is recomplemented (i.e., changes sign)
= 0 otherwise

T_2 = 1 if the result field is zero
= 0 otherwise

T_3 = 1 if $N_2 < \frac{1}{2}(N_1 + 1)$
= 0 otherwise

T_6 = 0 if $N_2 \leq 4$
= 1 otherwise

T_7 = 0 if $N_1 \leq 8$
= 1 otherwise

T_8 = 0 if fields do not overlap
= 1 otherwise

T_9 = 0 if any nonzero function byte is found
= 1 otherwise

T_{10} = 0 if $N_1 \geq N_2$
= 1 otherwise

T_{11} = 1 if $N_1 > \frac{1}{2}(N_2 + 1)$
= 0 otherwise

T_{13} = 0 if $N_2 \geq N_1$
= 1 otherwise

T_{14} = 1 if $NWBL_2 = 0$
= 0 otherwise

T_{16} = 0 if $N_1 \geq N_2$
= 1 otherwise

T_{18} = 1 if N = 1
= 0 otherwise

TB_1 = 1 if SBA > DBA
= 0 otherwise

TL_1 = 0.5 if $NWBL_1 > NWBL_2$
= 3.5 if $NWBL_1 < NWBL_2$

W = total number of double words in the first operand, and for those instructions with a single length

READER'S COMMENT FORM

IBM System/360 Model 65
Functional Characteristics

Form A22-6884-1

- Your comments, accompanied by answers to the following questions, help us produce better publications for your use. If your answer to a question is "No" or requires qualification, please explain in the space provided below. Comments and suggestions become the property of IBM.

	Yes	No
• Does this publication meet your needs?	☐	☐
• Did you find the material:		
Easy to read and understand?	☐	☐
Organized for convenient use?	☐	☐
Complete?	☐	☐
Well illustrated?	☐	☐
Written for your technical level?	☐	☐

- What is your occupation? _____
- How do you use this publication?

As an introduction to the subject?	☐	As an instructor in a class?	☐
For advanced knowledge of the subject?	☐	As a student in a class?	☐
For information about operating procedures?	☐	As a reference manual?	☐

Other _____

- Please give specific page and line references with your comments when appropriate. If you wish a reply, be sure to include your name and address.

COMMENTS:

- Thank you for your cooperation. No postage necessary if mailed in the U.S.A.

A22-6884-1

FOLD FOLD

FIRST CLASS
PERMIT NO. 116
KINGSTON, N. Y.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

POSTAGE WILL BE PAID BY
IBM CORPORATION
CPO BOX 120
KINGSTON, N. Y. 12401

ATTN: PRODUCT PUBLICATION MANUAL WRITING
DEPARTMENT 526

FOLD FOLD

IBM

International Business Machines Corporation
Field Engineering Division
112 East Post Road, White Plains, N.Y. 10601
[USA Only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]