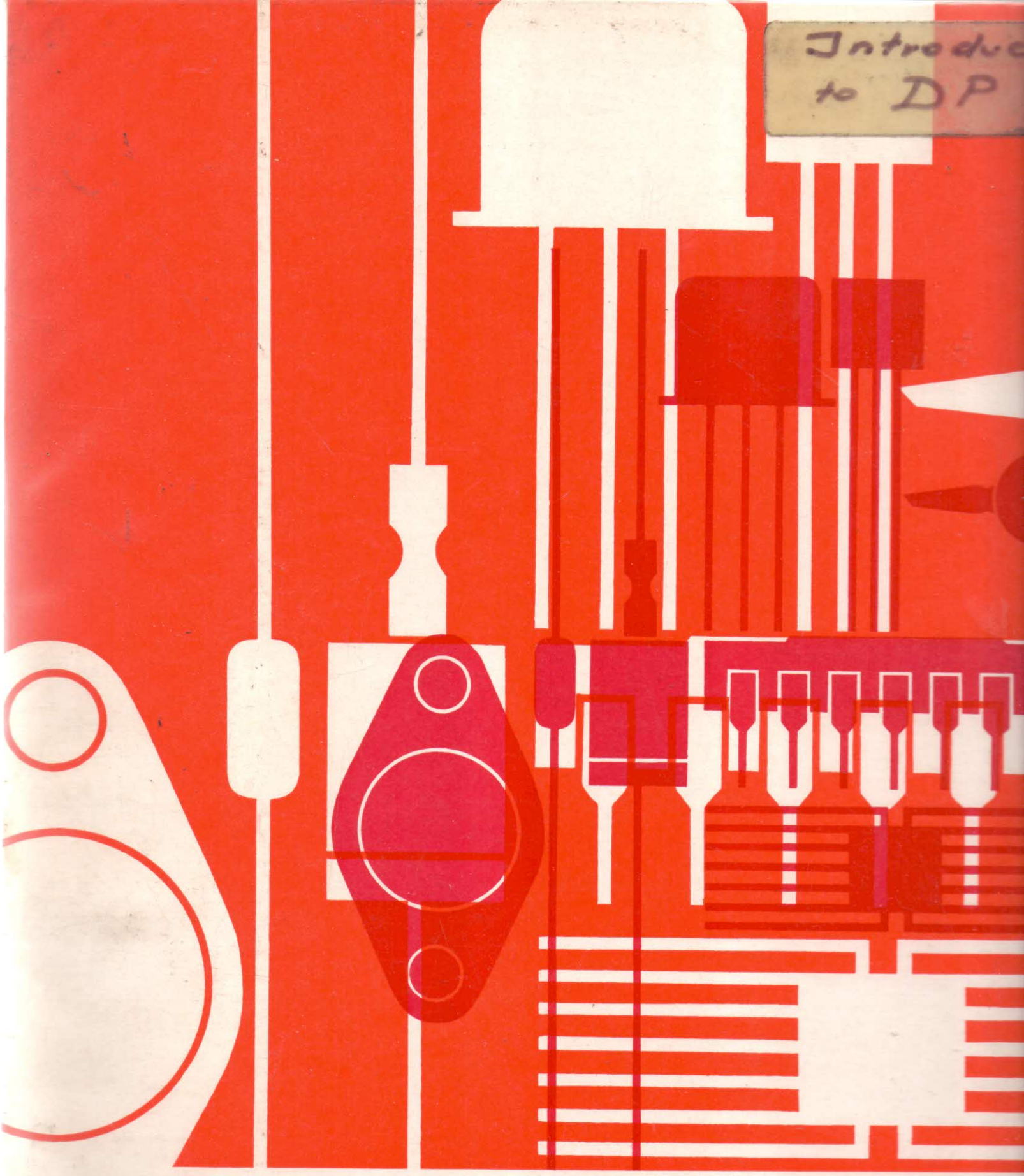


Introduc
to DP

W. Bellen

INTERMETALL Halbleiterbauelemente

File No. S360-01
GA22-6898-2

Systems Reference Library

IBM System/360 Model 50

Functional Characteristics

This manual presents the organization, characteristics, functions and features unique to the IBM System/360 Model 50. Major areas described are system structure, generalized information flow, standard and optional features, system control panel, instruction timings, channel characteristics, concurrent input/output capabilities, selector channel loading, multiplexer channel loading, and channel interference with the CPU.

Descriptions of specific input/output devices used with the IBM System/360 Model 50 appear in separate publications.

Configurators for the IBM 2050 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 Summary*, Form A22-6810.

Preface

Publications pertinent to the operation of System/360 Model 50 include:

IBM System/360 Special Feature Description — 7074 Compatibility Feature for System/360 Models 50 and 65, Form A27-2717

IBM System/360 Input/Output Configurator, Form A22-6823

IBM System/360 Data Communications and Acquisition Configurator, Form A22-6824

IBM System/360 Model 50 Operating Procedures, Form A22-6908

IBM System/360 Processors Models 44 and 50 (Equipment Template), Form X22-6914

Additional publications, categorized by type, include:

CHANNEL-TO-CHANNEL ADAPTER — *IBM System/360 Special Feature Channel-to-Channel Adapter*, Form A22-6892

COMPATIBILITY — *IBM System/360 Conversion Aids: The 1410/7010 Emulator Program for IBM System/360 Model 50*, Form C28-6568, and *IBM System/360 Conversion Aids: The 7074 Emulator Program for IBM System/360 Models 50 and 65*, Form C27-6908

CONSOLE TYPEWRITER — *IBM System/360 Component Descriptions and Operating Procedures — IBM 1052 Printer-Keyboard Model 7 with IBM 2150 Console*, Form A22-6877

DIRECT CONTROL — *IBM System/360 Direct Control and External Interrupt Features, OEMI*, Form A22-6845

SHARED 2361 STORAGE — *IBM 2361 Core Storage, OEMI*, Form A22-6869

POWER CONTROL INTERFACE *(for emergency power-off control)* — *IBM System/360 Power Control Interface*, Form A22-6906

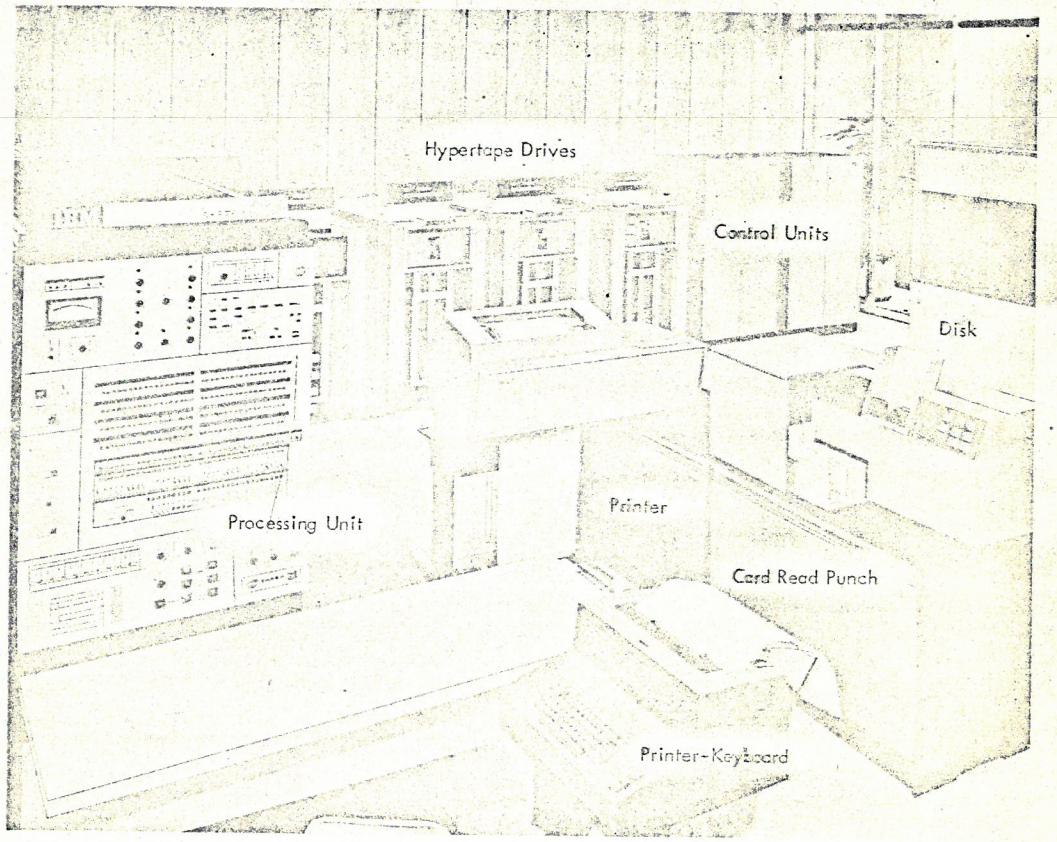
Third Edition (March, 1969)
This is a minor revision of, and obsoletes, Form A22-6898-1 and these related TNL's: N22-0277, N22-0292, N22-0299, N22-0309, and N22-0322.

Specifications contained herein are subject to change from time to time. Any such change will be reported in subsequent revisions or Technical Newsletters.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

This manual has been prepared by the IBM Systems Development Division, Product Publications, Dept. B98, PO Box 390, Poughkeepsie, N.Y. 12602. A form is provided at the back of this publication for readers' comments. If the form has been removed, comments may be sent to the above address.

Contents



IBM System/360 Model 50

The IBM System/360 Model 50 is part of a series of distinguished, compatible, high performance, data processing systems. The Model 50 provides the reliability, convenience, and confidence demanded by large-scale business and scientific computation, as well as general-purpose data processing for communications or control applications.

The Model 50 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. The high performance of its logical structure is principally due to:

1. Access to four bytes in parallel with a 2-microsecond processor storage cycle time, with a standard capacity ranging from 65,536 bytes for model F to 524,288 bytes for model I.
2. Local storage (with a 0.5-microsecond cycle time), used for general and floating-point registers, as well as a "scratch-pad" for various channel and CPU activities.
3. Read only storage (with a 0.5-microsecond cycle time) containing a microprogram that controls system operation.
4. Both arithmetic and storage operations are carried out using a full 32-bit word in parallel.
5. Overlap of channel (I/O) operations with CPU operations.

A partial list of options available to Model 50 includes (see "Preface" for list of associated publications):

Storage: Up to 8,388,608 bytes of 8-microsecond main storage are available under direct control of the processing unit.

Shared 2361 Storage: The IBM 2361 Core Storage may be shared with any Model 50, 65, or 75 that has equal or greater processor storage.

Direct Control: Permits control and synchronization between two CPU's, or between CPU and non-standard external device.

Compatibility: Permits system to emulate other systems, executing programs written for the other system. Either the 1410/7010 or the 7070/7074 compatibility feature may be plant installed in the Model 50.

Console Typewriter: The printer-keyboard is installed in operator position and includes program-controlled audible alarm.

Channel-to-Channel Adapter: Permits interconnection of two System/360 channels (one per Model 50).

Channels: Up to three selector channels. Additional multiplexer subchannels in models H, HG, or I expand the number of subchannels from 128 to 256. In the expanded version, all 256 subchannels are unshared.

The major component in a System/360 Model 50 is the 2050 Processing Unit, which contains arithmetic logic, main storage, read only storage, local storage, the standard multiplexer channel, and up to three selector channels. To this unit are attached auxiliary (optional) storage, input/output devices and their related control units (Figure 1).

The five models of the Model 50 are termed F50, G50, H50, HG50, and I50. These models differ only in the amount of main storage contained within the 2050 processing unit and the number of multiplexer subchannels. The significant differences are:

IBM SYSTEM/360 MODEL	PROCESSING UNIT MODEL	DESCRIPTION
F50	2050F	65,536 bytes of main storage 64 multiplexer subchannels
G50	2050G	131,072 bytes of main storage 128 multiplexer subchannels
H50	2050H	262,144 bytes of main storage 128 multiplexer subchannels
HG50	2050HG	393,216 bytes of main storage 128 multiplexer subchannels
I50	2050I.	524,288 bytes of main storage 128 multiplexer subchannels

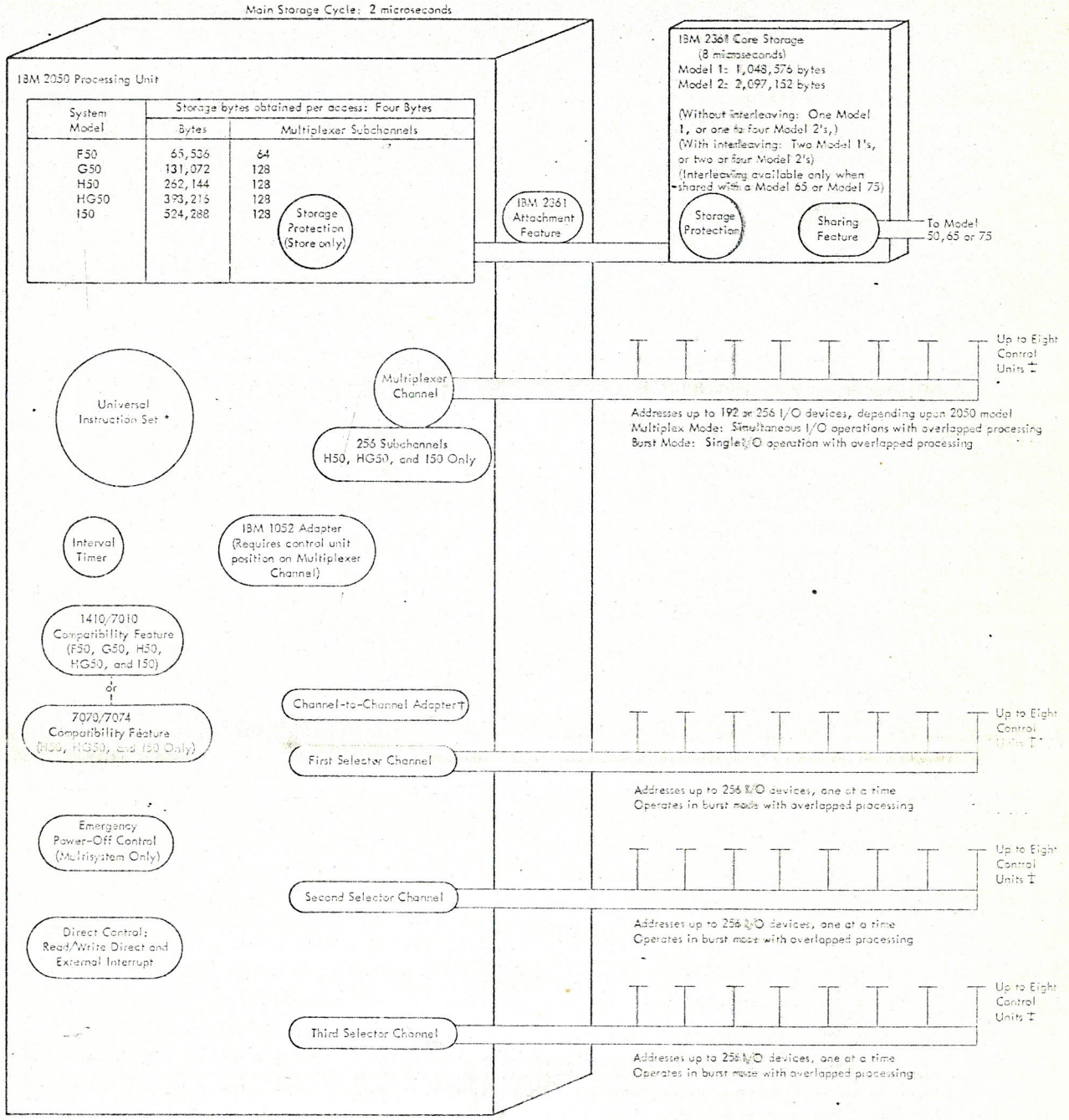
The F50 standard multiplexer channel has 64 subchannels, the G50, H50, and I50 have 128 subchannels. The number of subchannels can be increased to 256 on the H50, HG50, and I50 if desired.

The system control panel is located at one end of the 2050 Processing Unit. An optional 1052 Printer-Keyboard Model 7 may be mounted on the 2050 Processing Unit reading board to serve as an operator's console. (This is shown in Figure 1.) The operator control section of the system control panel may be duplicated once for each processing unit to provide a remote operator control panel, which can be mounted on either the 2150 Console, or the 2250 Display Unit Model 1. Standard features for any System/360 Model 50 include:

- Multiplexer channel
- Universal instruction set
- Interval timer
- Storage protection (store protection only)

Optional features for any Model 50 include:

- Direct control
- Selector channels (one, two, or three)
- Channel-to-channel adapter
- 1052 Adapter, 1052 Printer-Keyboard (Console Typewriter)



Color: Specify blue, yellow, red, or gray.

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 2050; Multiplexer or Selector) 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 a control unit.

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

Figure 1. IBM System/360 Model 50 Configurator

2361 Core Storage attachment
Multiplexer subchannels — additional (H50, HG50, and I50 only)
1410/7010 Compatibility feature (all models)
7070/7074 Compatibility feature (H50, HG50, and I50 only)

Outline configurations of the Model 50, produced by the various combinations of a 2050 Processing Unit, 2361 Core Storage, and various channel options are shown in Figure 1, including input/output devices attached to the channels through control units.

A variety of control units and input/output devices are available for use with the Model 50. 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.

Processing Unit

The 2050 Processing Unit contains the facilities for addressing main storage, for fetching or storing information, for arithmetic and logical processing of data, for sequencing instructions in the desired order, and for initiating the communication between storage and external devices. The four models of the 2050 Processor Unit vary only in the capacity of the main storage unit and in the number of multiplexer subchannels.

The 2050 Processing Unit contains the following major components:

Main storage
Arithmetic-logic unit
Local storage
General registers
Floating-point registers
Read only storage (ROS)
Multiplexer channel
System control panel
Optional selector channels

Main Storage

Main storage is available in the four storage capacities previously listed. The main storage read/write cycle time is 2 microseconds with access to four bytes. 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 installed main storage capacity.

The Model 50 transfers information between main storage and the processing unit in units of four bytes.

Main storage has a small extension, not accessible by the problem programmer, that is used to store the control and status information for each subchannel of the multiplexer channel. The number of these special storage locations determines the number of subchannels available to the multiplexer channel.

Arithmetic-Logic Unit

The arithmetic-logic unit contains a four-byte adder-subtractor that operates with either hexadecimal or decimal values. It is capable of producing both arithmetic and logical combinations of the input data streams. Cycle time is 0.5 microsecond.

Local Storage

Local storage consists of a small high-speed core storage unit providing registers for fixed-point and floating-point data, for channel operations, and for internal scratch-pad use. Local storage cycle time is 0.5 microsecond per four bytes.

General Registers

The 16 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 (four bytes). For some operations, two adjacent registers can be coupled together, providing a double word capacity. The general registers are implemented in local storage and have a cycle time of 0.5 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) floating-point operand. The floating-point registers are implemented in local storage and have a cycle time of 0.5 microsecond per four bytes.

Read Only Storage

The control function of the Model 50 is achieved by the use of a read only storage (ROS), which contains a permanent microprogram used to control the functions of data flow and instruction execution. Addressing ROS (except for the use of the DIAGNOSE instruction) is not available to the programmer.

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. When used with the Model 50, however, only four bytes are accessed in parallel to match the internal data flow of the system.

The 2361 is an extension of the main storage (processor storage) and is addressed contiguously with the 2050 processor storage. The 2361 Model 1 has a storage capacity of 1,048,576 bytes, and the Model 2 has a storage capacity of 2,097,152 bytes.

A Model 50 can share a 2361 with a System/360 Model 65, another Model 50, or a Model 75, if the other model has equal or greater processor storage.

When a 2361 is shared, its addresses are an extension of the addresses of the larger of the two processor storages.

Storage protection is a standard feature on the 2361 and matches the type of protection provided for the System/360 model that is using it. For use by the Model 50, the 2361 provides store protection only, but if another model with the fetch protection feature is sharing the 2361 with the Model 50, the 2361 provides both store and fetch protection for that other model. (The sharing of a common storage area by more than one program can be controlled by the use of the test and set instruction, which is described in the *IBM System/360 Principles of Operation*, Form A22-6821.)

The 2361's can be specified for two-way interleaving, when attached to a Model 65 or 75. Interleaving provides an addressing scheme between two 2361's that permits the overlapping of read/write storage cycles in sequential operations.

One 2361 Model 1 or one to four Model 2's, without interleaving, can be used with Model 50. Two 2361 Model 1's or two or four 2361 Model 2's, with interleaving, can be used with Model 50 when 2361's are shared with a Model 65 or Model 75. Any 2361's intended for intermixing with other 2361's must be equipped for two-way interleaving.

Channels

The channel directs the flow of information between the I/O devices and main storage. It 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 one byte at a time between the I/O device and the channel. Data transfers between the channel and storage are parallel by four bytes (one word) for selector channels. The multiplexer channel routes data to and from storage in units of a single byte (Figure 2).

For efficiency, the channels are integrated with the processing unit and share many of its facilities. For example, the channels utilize the same read only storage for control, and use the CPU data paths for handling nearly all data and control information. A standard I/O interface provides a uniform method of attaching I/O control units to all channels, making the Model 50 adaptable to a broad spectrum of applications and devices.

Channel-to-Channel Feature

A channel-to-channel adapter is available as an optional feature. The adapter permits communication between two System/360 channels, thus providing the capability for interconnection of two processing units. The adapter uses one control unit position on each of the two channels. This feature is required on only one of the two connected channels. Only one channel-to-channel adapter can be installed on a Model 50.

Multiplexer Channel

The multiplexer channel is a standard feature of the Model 50. This channel is capable of controlling several low to medium speed I/O units concurrently in multiplex mode or a single high-speed unit in burst mode.

The channel facility necessary to sustain an I/O operation with an I/O device is called a subchannel. The number of multiplexer subchannels is determined by the size of the main storage unit. (See Figure 2.) In the multiplex mode, the multiplexer channel sustains concurrent I/O operations on several subchannels. Bytes of data are interleaved and transmitted to or from the selected I/O devices and to or from the desired locations in main storage. A maximum of eight control units may be attached to the multiplexer channel.

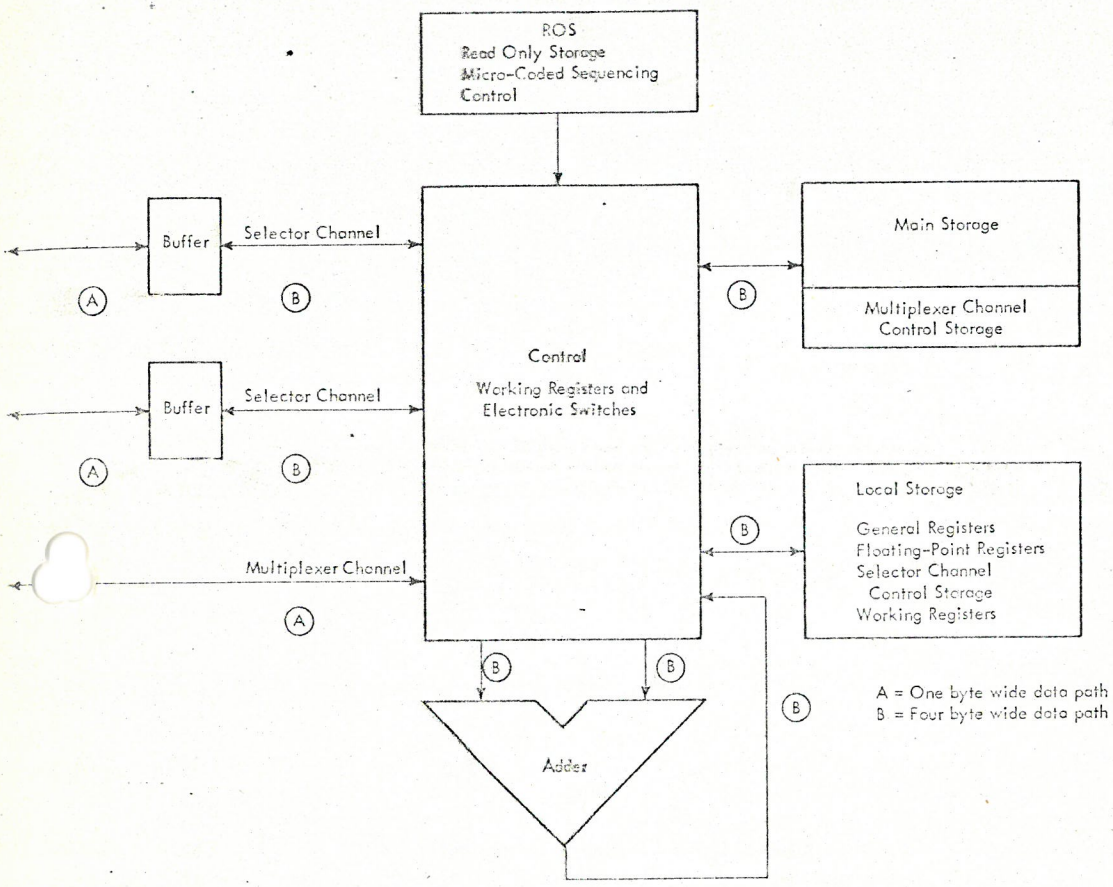
In the burst mode the multiplexer channel sustains one I/O operation on one subchannel. Only one I/O device can be selected at a time and no other device on the multiplexer channel can transfer data until the selected I/O activity has been terminated.

Selector Channel

One, two, or three selector channels are available, as optional features, for the Model 50. The selector channel operates in burst mode only, although one to eight control units can be attached and the channel has the facilities for addressing as many as 256 devices. Only one I/O device may be selected at a time on a selector channel. No other I/O device on the selector channel can transfer data until the selected activity has been terminated.

Control Panel

The control panel located on one end of the 2050 Processing Unit provides the switches, the keys and the lights necessary to operate, monitor and control the Model 50. The need for operator manipulation of manual controls is held to a minimum by the system design and the governing supervisory program. A detailed description of operator functions provided by the switches, keys and lights of the con-



	Capacity/Number	Data Width	Access/Speed/Rate
General registers	16	4 bytes	0.5 microsecond R/W cycle/4 bytes
Floating-point registers	4	8 bytes	0.5 microsecond R/W cycle/4 bytes
Adder		4 bytes	0.5 microsecond
Local storage			0.5 microsecond R/W cycle/4 bytes
Read only storage			0.5 microsecond Rd cycle
Main storage		4 bytes	2.0 microsecond R/W cycle
Basic machine cycle			0.5 microsecond
Multiplexer channel			
Burst mode		1 byte	
Multiplex mode		1 byte	
Data transfers			
Processor to storage		4 bytes	
Storage to storage		4 bytes	
Selector channel to storage		4 bytes	
Multiplexer channel to storage		1 byte	
Control unit to channel		1 byte	

Figure 2. Data Flow Diagram and System Statistics

trol panel is located in the section "System Control Panel."

Universal Instruction Set

The universal instruction set is a standard feature of the Model 50. It includes the standard, commercial, and scientific instruction subsets, plus the two storage protection instructions.

Descriptions of all instructions are found in the *IBM System/360 Principles of Operation*, Form A22-6821. Timing information for each of the instructions is found in "Instruction Times."

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.

System controls are logically divided into three classes: operator control, operator intervention, and customer engineer control (key switch and meters). This section of the manual discusses the system control functions provided by the system control panel as well as the purpose and use of the switches and lights on the panel.

System Control Functions

Using the control panel, the operator can perform these system control functions:

1. Reset the system.
2. Store and display information in storage and registers.
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. The parity of the general and floating-point registers, as well as the parity of the PSW, is corrected. All error-status indicators and the instruction address register are reset to zero.

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, when a PSW restart is performed, 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 the result registers or storage locations are unpredictable. If the CPU is in the wait state when the system reset is performed, and no I/O operation is in progress, this uncertainty is eliminated.

A system reset does not correct parity in storage but does correct parity in the registers. 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 direct storing and displaying of data. 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 and display keys, the address switches, the data switches and the storage select switch. Once the desired intervention is completed, the CPU can be started again.

The stopping and starting of the CPU in itself does not cause any alteration in program execution other than in the time element necessary for 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 masked off, 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 initiates a read operation from the selected input device. When reading is completed satisfactorily, the IPL PSW is obtained, the CPU starts operating, and the load light is turned off.

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 transferring data. The first 24 bytes read are placed in storage locations 0-23. Store protection, program controlled interruption, and a possible incorrect length indication are ignored. Control of the loading operation is then assumed by the double word just read into location 8, which is used as the next channel command word (CCW). The remainder of the program to be loaded may therefore be located in any desired section of storage. When chaining is specified in this CCW, the double word in location 16 may also be used as a CCW to provide additional control.

After the input operation is completed, 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 then fetches the double word in location 0 as a new PSW and proceeds as in normal operation. 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 simulated by CPU circuitry, 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.

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

The selected input device may be a channel-to-channel adapter connecting the channels of two CPU's. After a system reset is performed and a read command is issued to the adapter by the requesting CPU, the adapter sends an attention signal 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, operator intervention, and customer engineer control. Figure 3 shows the operator controls located in areas labeled C and N and operator intervention controls in areas L, M, and N of the system control panel. The customer engineer will use all controls, but the controls in areas A, B, F, G, and K are intended primarily for customer engineer use.

Operator Controls

Sections C and N 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 similar to 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 following table lists (alphabetically) all operator controls and indicators and their implementation. All operator controls except the emergency pull switch are located in the area of the control panel labeled N and shown in Figure 4. The controls in area N are identical in all models of the System/360. The emergency pull switch is located in area C.

NAME	IMPLEMENTATION
Emergency Pull	Pull switch
Interrupt	Key
Load	Key
Load	Light
Load Unit	Three rotary switches
Manual	Light
Power Off	Key
Power On	Key, backlighted
System	Light
Test	Light
Wait	Light

Note: All keys have momentary pushbutton action.

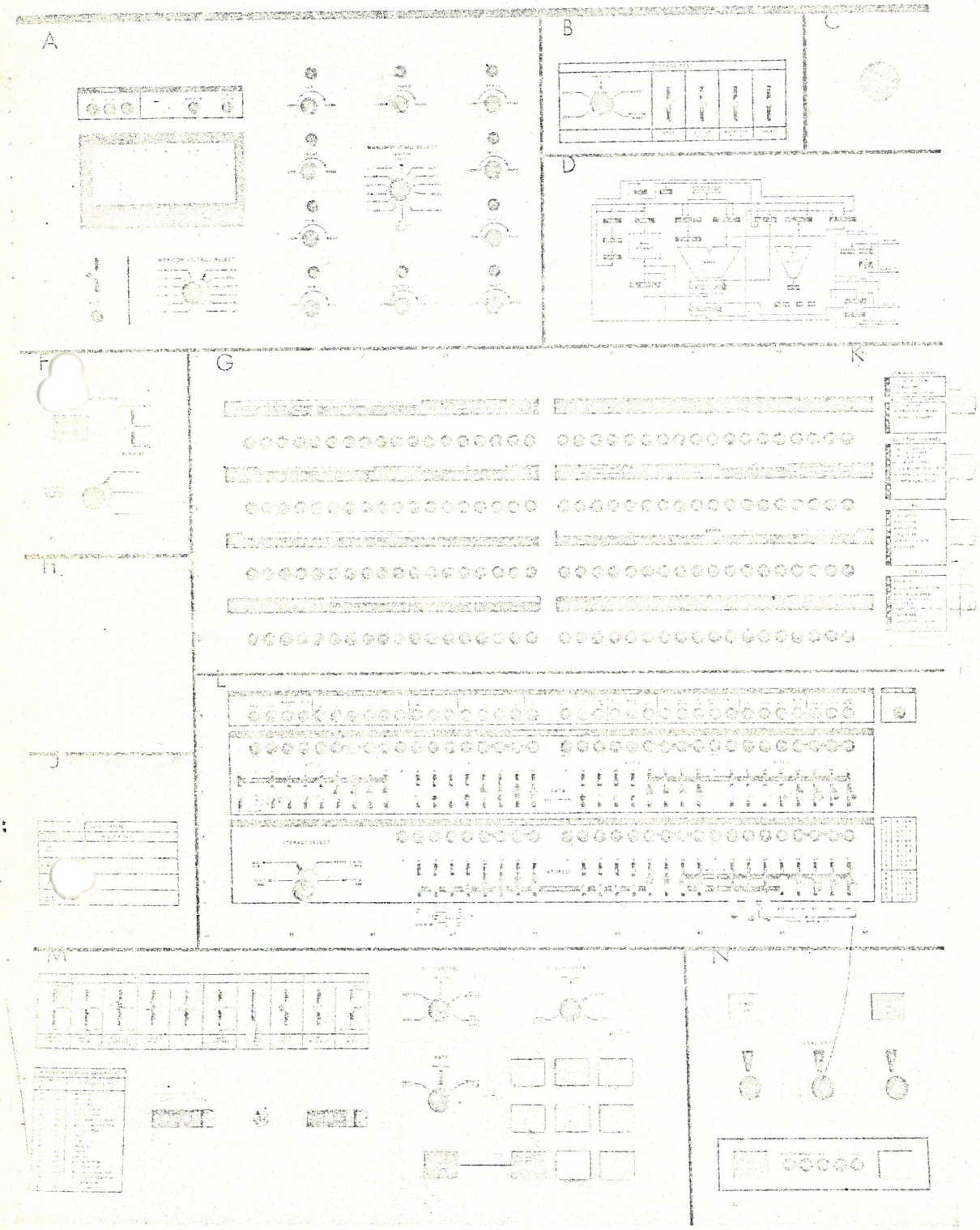


Figure 3. System Control Panel

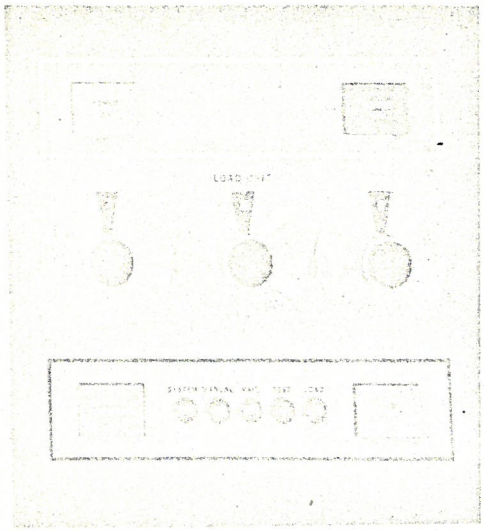


Figure 4. Section N Panel

Emergency Pull

Pulling this switch turns off all power beyond the power-entry terminal on every unit that is part of the system or that can be switched onto the system.

The switch latches in the out position and can be restored to its normal position by maintenance personnel only.

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

Interrupt

The interrupt key is pressed to request an external interruption.

The interruption is taken when not masked and when the CPU is not stopped. Otherwise, the interruption request remains pending. Bit 25 in the interruption-code portion of the current PSW is made 1 to indicate that the interrupt key is the source of the external interruption. The key is effective while power is on the system.

Load (Key)

The load key is pressed to start initial program loading. The key is effective while power is on the system.

Load (Light)

The load light is on during initial program loading; it is turned on when the load key is pressed and is turned off after the read operation and the loading of the new PSW are completed successfully.

Load Unit

Three rotary switches provide the 11-bit address of the channel and unit to be used for initial program loading.

The leftmost rotary switch has eight positions labeled 0-7 used for the channel address. The other two 16-position rotary switches are labeled with the hexadecimal characters 0-9, A-F, and are used for the unit address.

Manual

The manual light is on when the CPU is in the stopped state. Several of the manual controls are effective only when the CPU is stopped (manual light on).

Power Off

The power-off key is pressed to initiate the power-off sequence of the system.

The contents of main storage (but not the keys in storage associated with the protection feature) are preserved, provided that the CPU is in the stopped state. The key is effective while power is on the system.

Power On

This key is pressed to initiate the power-on sequence of the system.

As part of the power-on sequence, a system reset is performed in such a manner that the system performs no instructions or I/O operations until explicitly directed. The contents of main storage are preserved.

The power-on key is backlighted to indicate when the power-on sequence is completed. The key is effective only when the emergency pull switch is in the normal position.

System

The system light is on when the CPU usage meter or customer engineer meter is running.

Test

The test light is on when a manual control is not in its normal position or when a maintenance function is being performed for CPU, channels, or storage. The normal position for rotary switches is straight up, and for lever switches is straight out.

Any abnormal switch setting on the system control panel or on any separate maintenance panel for the CPU, storage, or channels that can affect the normal operation of a program causes the test light to be on.

The test switches described are shown in Figures 5, 6, and 7. These switches cause the test light to be on if any is not in its normal position:

SWITCHES	PANEL
Rate switch not to process	M
FLT Mode switch not to off	M
Address Compare switch (IAR) not to process	M
Address Compare switch (ROS) not to sync	M
Repeat INSN on (IAR or ROS) not straight out	M
Check Control not to process	M
Storage Test not to process	B

SWITCHES	PANEL
FLT Control not to process	M
Lamp Test not straight out	M
Rev Data Pty not straight out	L
Disable Interval Timer not straight out	M
Invert SAR Bit 16 not straight out	B
Meter switch to CE	M
Manual Op not straight out	F
SAR Compare to stop	M

The test light may be on when one or more diagnostic functions under control of DIAGNOSE is activated or when certain abnormal circuit breaker or thermal conditions occur.

The test light does not reflect the state of marginal voltage controls.

Wait

The wait light is on when the CPU is in the wait state.

Processing Notes

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 states of the wait and manual lights. The possible conditions when power is on are:

SYSTEM LIGHT	MANUAL LIGHT	WAIT LIGHT	CPU STATE	I/O STATE
Off	Off	Off	*	*
Off	Off	On	Wait	Not working
Off	On	Off	Stopped	Not working
Off	On	On	Stopped, Wait	Not working
On	Off	Off	Running	Undetermined
On	Off	On	Wait	Working
On	On	Off	Stopped	Working
On	On	On	Stopped, Wait	Working

* Abnormal condition

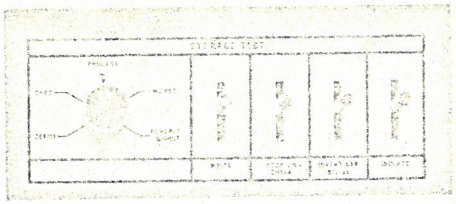


Figure 6. Section B Panel

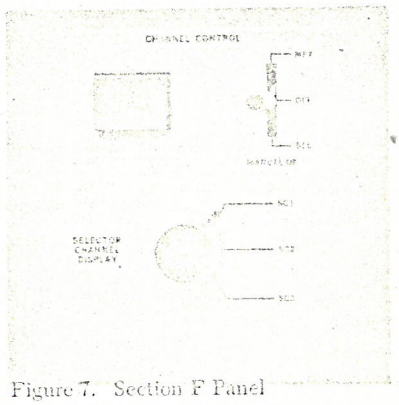


Figure 7. Section F Panel

Operator Intervention Controls

Sections L, M, and N of the system control panel contain the controls required for the operator to intervene in normal programmed operation. These controls are intermixed with the customer engineer controls. Only operator intervention controls are described in detail.

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

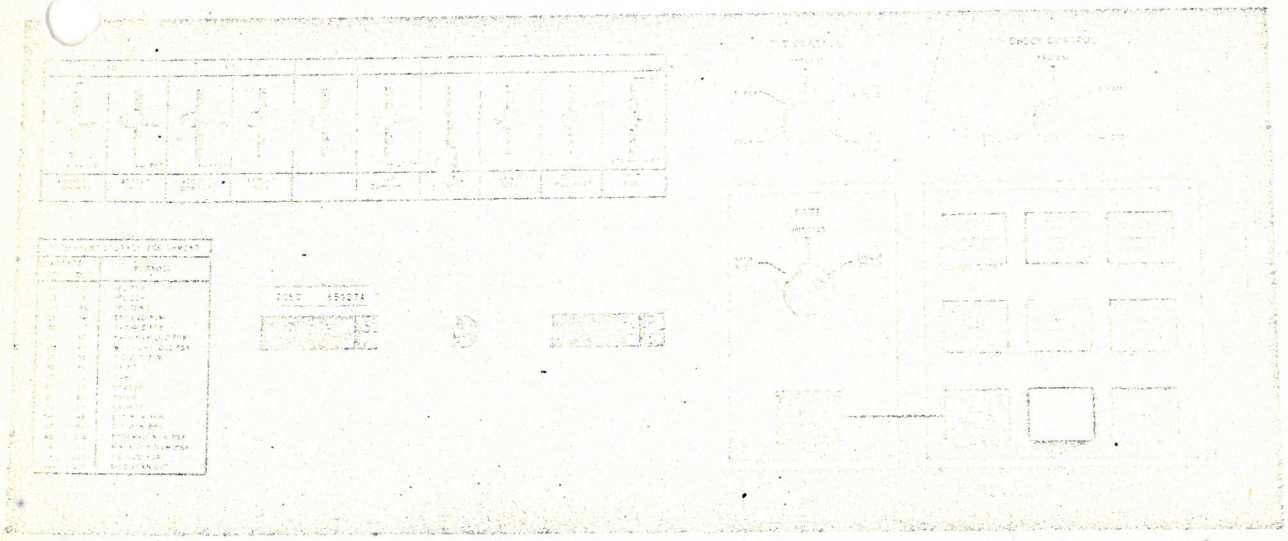


Figure 5. Section M Panel

The following table lists (alphabetically) all intervention controls and indicators and their implementation. Keys have momentary pushbutton action. The section L panel is shown in Figure 8.

NAME	PANEL	IMPLEMENTATION
Address	L	Key switches
Address Compare (IAR)	M	Key switch
Check Control	M	Rotary switch
Check Reset	M	Key
Data	L	Key switches
Display	M	Key
PSW Restart	M	Key
Rate	M	Rotary switch
Rev Data Pty	L	Key switch
SAR Compare	M	Key switch
Set IC	M	Key
Start	M	Key
Storage Select	L	Rotary switch
Store	M	Key
Stop	M	Key
System Reset	M	Key

Address

The address lever switches provide a means of manually selecting an addressable location in storage when used in conjunction with the storage select switch or to identify the address to be compared when an address-comparison stop is desired. Correct parity is automatically generated.

For main storage select, the address switches are used to manually address a main storage location when the storage select switch is set to the MAIN position. For an address-comparison stop, the address switches provide the stop address. When addressing main storage, the 24 switches represent a 24-bit binary address. The rightmost toggle is the units position or low-order position. Because data in main storage are stored or displayed a word at a time, the two low-order position address switches (switch positions 30 and 31) are not involved in determining the address.

When an address switch is in the down position it represents a one bit; when in the center or restored position it represents a zero bit. Color coding is provided to identify the hexadecimal digit groupings.

If the address switches are manipulated while address compare (IAR) is set to stop, a machine check can occur.

Address Compare (IAR)

The address compare switch provides a means of stopping the CPU on a successful instruction address comparison.

1. In the stop position, an equal comparison between the address switches and the instruction address register causes the CPU to stop. The stop occurs at completion of the addressed instruction.
2. In the process position, no comparison occurs.
3. The sync position is for customer engineer use.

The address compare switch can be manipulated without disrupting CPU operation other than by causing the address-comparison stop. When the switch is set to sync or stop position, approximately 3 microseconds are added to each instruction to accomplish the comparison. To avoid a machine check interruption, set the address switches and then set the address compare switch to the stop position.

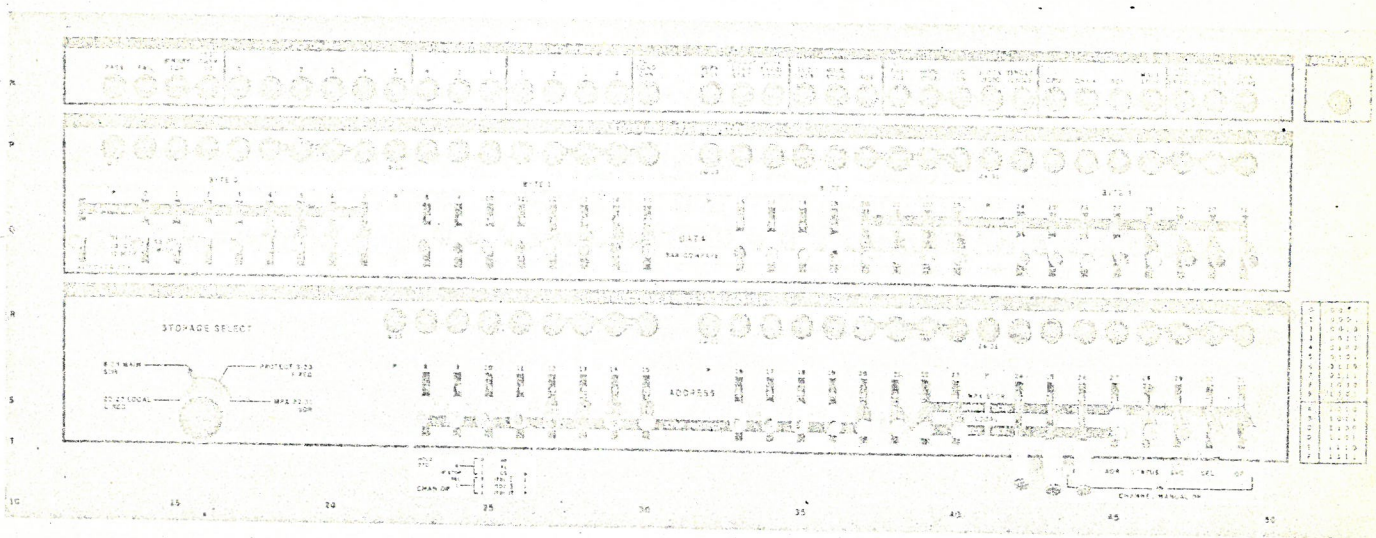


Figure 8. Section L Panel

Check Control

The check control switch checks for errors in the CPU/channel operation. This switch has the following positions:

1. *PROCESS (Normal Position):* In this position, all errors are handled by the monitor with the aid of a general log-out, if PSW bit 13 is unmasked. This is followed by an interruption. If PSW bit 13 is masked, the error register is set, but the error remains pending.
2. *DISABLE:* In this position, all errors in the CPU or multiplexer channel are ignored and the system operation continues, disregarding the error. The program operation can still be affected by the error. Errors in the selector channel (except data parity) cause an I/O interruption. (This position is generally used by the customer engineer.)
3. *CPU STOP:* In this position, processing is stopped when a machine check interruption occurs; the CPU is also stopped (hardstop).
4. *CHAN STOP:* In this position, processing is stopped when an error other than incorrect-length indication (ILI) occurs. (This position is generally used by the customer engineer.)

Check Reset

The check reset key resets all CPU and channel check indicators to the no-error state. Check reset can be considered a subset of the system reset. It is active in all modes. Check lights remaining on after check reset must be cleared at the check source by use of appropriate manual controls.

Data

The storage data switches in section L are used to specify or represent the data to be stored in the location specified by the storage select switch and address switches. Correct data parity is automatically generated. Changing the keys does not affect CPU operation.

The storage data switches can be used to represent a word of information. The leftmost switch is not used as part of the word for customer store functions. A storage data switch in the down position represents a one bit and in the center position a zero bit.

Display

The display key is pressed to display information that is in the location specified by the storage select switch and address switches. When the designated location is not available, the displayed information is unpredictable. The key is effective only while the CPU is in the stopped state.

PSW Restart

This key causes a system reset, after which a PSW is loaded from storage location zero and the CPU is changed from stopped to operating state.

Rate

This three-position rotary switch is used to indicate the manner in which instructions are to be performed. The position of the switch should be changed only while the CPU is in the manual state. Otherwise unpredictable results may occur. The rate switch has the following settings:

1. *PROCESS:* In this position, the system starts operating at normal speed when the start key is pressed. The test light is on when the rate switch is not set to process position. Moving the rate switch from process to instruction-step position stops the CPU.
2. *INSN STEP (Instruction Position):* In this position, the system executes one instruction for each depression of the start pushbutton and returns to the manual state. All pending interruptions not masked are subsequently taken. The timer is not updated when the switch is in this position.

   Any instruction can be executed with the rate switch set to INSN STEP. Input/output operations are completed to the interruption point. When the CPU is in the wait state, no instruction is performed, but pending interruptions, if any, are taken before the CPU returns to the stopped state. Initial program loading is completed with the loading of the new PSW before any instruction is performed.
3. *SINGLE CYCLE:* The system executes one machine cycle for each depression of the start pushbutton and returns to the stopped state. The stopped state for single cycle is one in which no CPU clocks are running. Otherwise, in the normal stopped state, the ROS is running, executing a halt loop.

   Single cycle operates with I/O equipment to the point of the initiation of the asynchronous operation. The asynchronous operation starts with the next depression of the start pushbutton and runs to completion. If start pushbutton is pressed during this time, the next cycle is taken. If an interruption results, the interruption sequence is not automatically executed but must be single cycled. Moving the rate switch from process to single-cycle position while the CPU is running stops the CPU.

Rev Data Pty

This switch generates incorrect parity for data specified in the data switches. Parity is inverted for all bytes of the word. When this switch is on, the test light is on.

SAR Compare

The storage address register compare switch provides a means of stopping the CPU on a successful data address comparison.

1. In the stop position, an equal comparison between the data lever switches 8-31 (low-order 24 positions) and a storage address may be used either to locate data or as a successful branch address. Either comparison causes the CPU to stop at the completion of the instruction containing that address. Data switch positions 0 and 1 may be set to effect the stop when the storage reference is made either by the CPU, by the channels, or by both.
2. In the sync position, no stop occurs.

The SAR compare switch can be manipulated without disrupting CPU operation other than by causing the address-comparison stop.

Set IC (Instruction Counter)

This key enters the instruction address set in the address switches. The key is active only when the CPU is in the stopped state.

Start

The start key is pressed to start instruction execution as specified by the rate switch. The key is effective only while the CPU is in the stopped state.

Pressing the start key after a normal stop causes instruction processing to continue as if no stop had occurred, provided that the rate switch is in the process or instruction-step position. If the start key is pressed after a system reset without introducing a new instruction address (SET IC), the results are unpredictable.

Storage Select

This four-position rotary switch is used to select the main storage area addressed by the address switches. The storage select switch is active only in stopped state (manual light on). The storage select switch has the following settings:

1. *MAIN (Main Storage):* Selects a main storage location specified by the address switches.
2. *PROTECT (Storage Protect):* Unconditionally selects the storage protect key register.
3. *LOCAL:* Selects a local storage location (PSW, general or floating-point register, and working storage) specified by the address switches.
4. *MPX:* Used by the customer engineer.

The switch can be changed without disrupting CPU operations.

Store

The store key is pressed to store information in the location specified by the storage select switch and address switches. The key is effective only while the CPU is in the stopped state.

The contents of the data switches are placed in specified locations in main storage, in general registers, or in floating-point registers. Store protection is ignored. When the locations designated by the address switches and storage switch are not available, data are not stored.

Stop

The stop key causes the CPU to enter the stopped state and turns on the manual light. (The CPU first completes the instruction being executed at the time the stop signal is recognized, and it processes all pending unmasked interruptions.) While the CPU is in the manual (stopped) state, any I/O operation in progress is completed. The key is effective while power is on the system.

Pressing the stop key has no effect when a continuous string of interruptions is performed or when the CPU is unable to complete an instruction because of machine malfunction.

System Reset

This key resets the CPU, channels, and control units, to initial state. The CPU is placed in the manual state, all pending interruptions are eliminated and all error indicators are reset. The key is effective while power is on the system.

Key Switch and Meters

The customer usage (CPU) meter and the customer engineer meter are on panel M of the system control panel. The customer engineer *key* switch controls which of these meters is to run while the system is in operation; that is, initiating, executing, or completing instructions, including I/O and assignable unit operations. The test light is turned on when the key (meter) switch is in the customer engineer position. (For other conditions, see "Test.") The system light, on panel N, indicates when the system is in operation.

General Channel Information

IBM System/360 channels transfer data between core storage and I/O devices under control of a channel program executed independently of the CPU program. The Model 50 CPU is free to resume the CPU program after initiating an I/O operation.

Model 50 channels may run concurrently, within the data transfer rate and channel programming conventions specified in this manual.

A major feature of the channels is their common I/O interface connection to all System/360 input/output control units. The I/O interface provides for attachment of a variety of I/O devices to a channel.

At the end of an I/O operation, the channel signals an I/O interruption request to the CPU. If not masked off, an I/O interruption occurs that places the I/O new PSW in control of the CPU. When I/O interruptions are masked, interruption requests are queued. Until honored, an I/O interruption condition is called a pending I/O interruption.

At the end of an I/O operation, a channel has information concerning the success of the operation, or details about any lack of success. The information is available to the CPU program.

Each System/360 channel has facilities for performing the following functions:

Accepting an I/O instruction from the CPU
Addressing the device specified by an I/O instruction
Fetching the channel program from core storage
Decoding the channel command words that make up the channel program
Testing each channel command word (CCW) for validity
Executing CCW functions
Placing control signals on the I/O interface
Accepting control-response signals from the I/O interface
Transferring data between an I/O device and core storage
Checking parity of bytes transferred
Counting the number of bytes transferred
Accepting status information from I/O devices
Maintaining channel-status information
Signaling interruption requests to the CPU
Sequencing interruption requests from I/O devices
Sending status information to location 64 when an interruption occurs
Sending status information to location 64 upon CPU request

Channel Control

IBM System/360 channels provide a common input/output interface to all System/360 control units. All control units are governed with six basic channel commands and a common set of only four CPU instructions. The instructions are:

Start I/O
Test channel
Test I/O
Halt I/O

All I/O instructions set the PSW condition code, and, under certain conditions, all but test channel may cause a channel status word to be stored. A test channel instruction elicits information about the addressed channel; a test I/O instruction elicits information about a channel and a particular device. Halt I/O terminates any operation on the addressed channel, subchannel, or device. Only Start I/O makes use of channel command words (CCW's).

A start I/O instruction initiates execution of one or more I/O operations. It specifies a channel, subchannel, control unit, and I/O device. It causes the channel to fetch the channel address word (CAW) from location 72. The CAW contains the protection key and the address of the first channel command word (CCW) for the operation. The channel fetches and executes one or more CCW's, beginning with the first CCW specified by the CAW.

Six channel commands are used:

Read
Write
Read backward
Control
Sense
Transfer in channel

The first three are self-explanatory.

Control commands specify such operations as set tape density, rewind tape, advance paper in a printer.

A sense command brings information from a control unit into main storage concerning unusual conditions detected during the last I/O operation, and detailed status about the device.

A transfer in channel (TIC) command specifies the location in main storage from which the next CCW in the channel program is to be fetched. A TIC may not specify another TIC. Also, the CAW may not address a TIC.

Each CCW specifies the channel operation to be performed and, for data transfer operations, specifies contiguous locations in main storage to be used. One or more CCW's make up a channel program that directs a channel operation.

Channel Registers

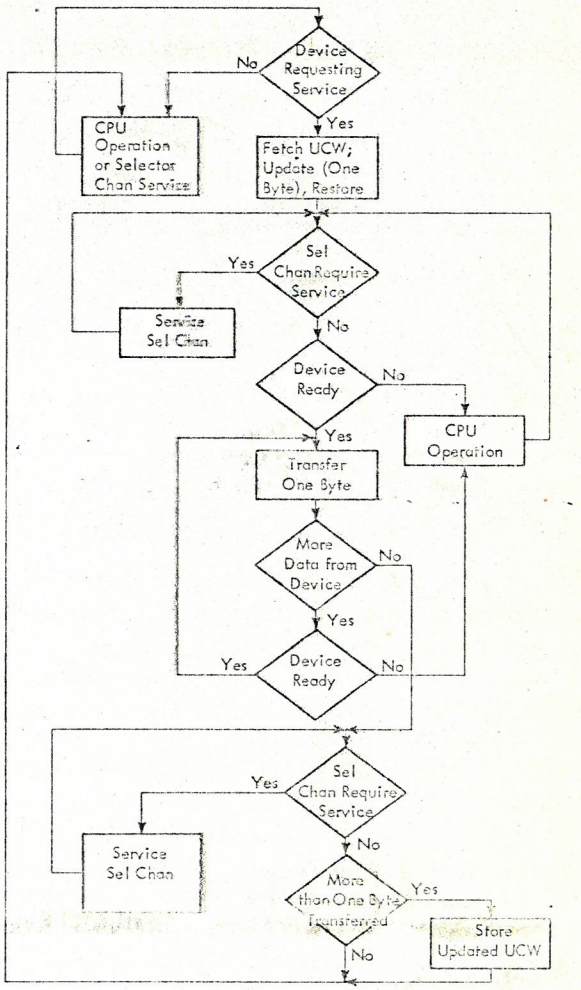
System/360 channels maintain the following channel control information for each I/O device selected:

Protection key
Data address
Identity of operation specified by command code
CCW flags
Byte count
Channel status
Address of next CCW

A selector channel has only one set of registers for the above information because it operates with only one I/O device at a time.

On a multiplexer channel, the listed information must be maintained for each subchannel in operation. Storage for this information is provided by special channel storage that is not directly addressable. Each subchannel has provision in channel storage for channel register information. When a particular subchannel is selected by a start I/O instruction and a channel program initiated, the channel storage locations for the subchannel are loaded with the information necessary for operation of the subchannel.

At each cessation of activity in a subchannel, its particular area in channel storage contains updated information, and the multiplexer channel is available for operation of another subchannel. The sharing of facilities by the multiplexer channel and the CPU is shown in Figure 9.



Note: During multiplexer operation, the selector channels are scanned for higher priority data service requirements as shown. The CPU is allowed to absorb available time due to delays in device responses.

Figure 9. Equipment Sharing by Model 50 CPU and Multiplexer Channel

Chaining

A single CCW may specify contiguous locations in main storage for a data transfer operation, or successive CCW's may be chained together to specify a set of noncontiguous storage areas. Chaining to the next CCW is caused by the presence of a flag bit in a CCW.

In data chaining, the address and count information in a new CCW is used; the command code field is ignored.

Entire CCW's, including their command code fields, may also be chained together for use in a sequence of channel operations. Such coupling is called command chaining, and is specified by a different flag bit in a CCW.

Data chaining has no effect on a device, as long as the channel has sufficient time to perform both data chaining and data transfer for the device.

In this manual, when a device is said to data chain, it means that the channel program for the device specifies data chaining.

Fetching Channel Command Words

The channel must fetch a new CCW whenever a CCW specifies data chaining, command chaining, or transfer in channel (TIC). The extra control activity caused by these operations takes time and diminishes a channel's ability to do other work.

A data chaining fetch usually occurs while a channel also has a data transfer load from the same device. The time required to fetch the new CCW necessarily limits the interval of time available for successive data transfers through the channel. An absence of data chaining ordinarily permits a channel to operate with a faster I/O device. Similarly, when a channel is not transferring data, a data chaining operation has a lesser impact on channel facilities.

Data Chaining in Gaps

For direct access storage devices, such as an IBM 2311 Disk Storage or an IBM 2314 Direct Access Storage Facility, formatting write commands cause the control

unit to create gaps between count, key, and data fields on the recording track. Read, write, and search commands that address more than one of the fields may specify data chaining to define separate areas in main storage for the fields.

The gaps on a track have significance to channel programming considerations for direct access storage devices. The channel does not transfer data during the time a gap is created or passes under the read-write head, and this time is sufficient for a Model 50 to perform a command chaining or data chaining operation.

Command chaining ordinarily occurs only during gap time, but data chaining may occur during gap time or while data is being transferred. A data chaining operation occurring during gap time has a lesser impact on channel facilities than when data transfers also occur. If a channel program for a direct access storage device calls for data chaining only during gap time, the device's overall load on channel facilities is significantly less.

When a direct access device is said to data chain in a gap, the reference is to a gap other than a gap following a data field. The latter gap causes a device end indication and command chaining is used in such a gap if the transfer of more information is desired. A device end occurring in the absence of a CCW specifying command chaining results in termination of the operation. When command chaining continues the operation, the status information available at the end of the operation relates to the last operation in the chain.

While reading, an attempt to data chain in a gap following a data field causes an incorrect length indication in the channel status byte.

Latency Command Chaining

Operation of direct access devices, such as disk storage, requires the use of command chaining. Between certain operations, such as the search for a record identification key and the reading of a data field on a direct access storage device, the control unit has a fixed time interval during which it must receive and execute a new command. If activity on other channel(s) causes too much delay in initiation of the operation specified by the new command, the channel program is terminated and an I/O interruption condition occurs.

Storage Addressing

When data chaining, the beginning and ending byte addresses and the minimum number of bytes transferred are factors in the maximum data rates that different System/360 channels can sustain. If the storage width of larger models and the possibility of using faster I/O devices are kept in mind when writing channel programs for small models, better performance will be obtained when the programs are run on larger models or with faster I/O devices.

For example, a tape operation at a 30 kb (kilobyte) data rate may data chain with a byte count of one on a Model 30 with one selector channel, but the same tape operation cannot be performed at 90 kb on a Model 50. In this instance, the use of a larger count for data chaining would permit the Model 50 to execute the channel program at 90 kb.

Channel Implementation

The Model 50 has two types of channels. The multiplexer channel is standard; up to three selector channels are optional. All channels on the Model 50 are integrated with the 2050 Processor and share part of the CPU facilities. Each channel may attach as many as eight control units and can address up to 256 I/O devices. Control units are connected to all channels through a standardized I/O interface.

Selector Channel

Each selector channel provides a path for moving data between storage and a selected I/O device. It has its own registers for control information and data buffering. Data move to or from an I/O device one byte at a time, but are buffered to a width of four bytes for communication with storage. Selector channels can operate concurrently with each other and with the CPU.

Multiplexer Channel

A multiplexer channel has a single data path that may be monopolized by one I/O device (burst mode) or shared by many I/O devices (multiplex mode). The design of a control unit predetermines whether its operation on the multiplexer channel is in burst or multiplex mode. In either case, data transfer between storage and an I/O device is controlled one byte at a time. Multiplexer channel operation may overlap selector channel and CPU operation.

When multiple I/O devices concurrently share multiplexer channel facilities, the operations are in multiplex mode. Each device in operation is selected, one at a time, for transfer of a byte or a few bytes to or from main storage. Bytes from multiple devices are interleaved together and routed to or from desired locations in core storage. Thus, the multiplexer channel data path is used by one device for transfer of one or a few bytes of data and then another device uses the same data path. The sharing of the data path

makes each device appear to the programmer as if it has a data path of its own. This leads to calling a device's share of the data path a subchannel.

The numbering scheme for multiplexer subchannels relates to I/O device addresses; the device address assigned to each device determines the subchannel that controls its operation. For an unshared subchannel, one device address is used. A shared subchannel permits use of several device addresses. The devices share a single control unit, which connects them to the shared subchannel. The devices may be selected for use one at a time, but may not be selected concurrently.

Device addresses 0 to 127 refer to correspondingly numbered unshared subchannels. Device addresses 128-255 are assigned to shared subchannels 0-7 in eight groups of 16. Physically, the shared subchannels are the same as the first eight nonshared subchannels. For example, the set of addresses 1000xxxx (128-143) refers to the same subchannel as the address 00000000 (000); the set 1001xxxx (144-159) refers to the same subchannel as the address 00000000 (001), etc.; the set 1111xxxx (240-255) refers to the same subchannel as the address 00000111 (007). Thus the installation of all eight sets of devices on the shared subchannels reduces the maximum possible number of devices that do not share a subchannel to 120.

The storage sizes on the Model 50 and the associated subchannels are:

MODEL	SUB-CHANNELS	I/O DEVICE ADDRESSES	SHARED SUBCHANNEL I/O DEVICE ADDRESSES	UNSHARED SUBCHANNEL I/O DEVICE ADDRESSES
F50	64	0-63,128-255	0-7,128-255	8-63
G50	128	0-255	0-7,128-255	8-127
H50	128	0-255	0-7,128-255	8-127
HG50	128	0-255	0-7,128-255	8-127
I50	128	0-255	0-7,128-255	8-127
H50*	256	0-255	none	0-255
HG50*	256	0-255	none	0-255
I50*	256	0-255	none	0-255

*For the H50, HG50, and I50, the additional multiplexer subchannels optional feature provides a total of 256 subchannels. Each is uniquely addressed; none of the 256 subchannels can be shared.

Channel Priority

Priority for allocation of Model 50 CPU facilities is in this order, for normal operation:

Machine check interruption handling
Multiplexer channel data transfer
Selector channel data transfer
Selector channel chaining
Multiplexer channel chaining
CPU operation

Selector channels receive priority in numeric order.

Although the multiplexer channel normally has higher priority than the selector channel, in certain cases, the selector channel may be promoted above the multiplexer channel. This occurs when selector channel service has been delayed to the point where it becomes likely that overrun will occur if the channel is not serviced immediately.

Relationship of Model 50 to Other Models of System/360

Model-Dependent Functions

The compatibility rule of the System/360 does not apply to a number of detail functions for which neither the frequency of occurrence nor usefulness of results warrants identical action on all models. These functions are concerned with the handling of invalid programs and machine malfunctions and are explicitly identified in the *IBM System/360 Principles of Operation,* Form A22-6821. Whenever model dependency exists, the definition of the System/360 allows choice in implementation or specifies that the operation is unpredictable. The user should ignore results that are defined to be unpredictable and should not base his program on any function where choice in implementation is permitted.

Considering any particular installation and operation, the operation normally is not truly unpredictable; the action may depend on the particular system components or on the input data. This section describes how some of the model-dependent functions are performed on the Model 50.

Note, however, that writing a program based on information in this section is in violation of the rules of compatibility of the System/360. If a program depends on a function that is model-dependent, it may not run on another model of the system. Even if the program includes the model-dependent operation of all other models of the System/360, difficulties may occur when new models are introduced. Furthermore, a mandatory engineering change may in some instances require a change in the execution of a model-dependent function in an installed machine, and therefore may require changes in a program that uses model-dependent information.

Logout

Logout of main storage starts at main storage location 128 (hex 80) and extends, for 164 bytes, to 292 (hex 120).

PSW Display

The low-order portion of the PSW is in local storage working register 7. This data can be stored or displayed by using the store or display function and by addressing local storage working register 7 (010111).

Program Interruption

Under the conditions of program interruption, the Model 50 does not cause an ILC of zero to be stored.

Controls and Indicators

Address-Compare Stop

The Model 50, in addition to the conventional address-compare stop using the instruction address register as reference, also provides a stop when the address in SAR is reached.

The SAR compare switch compares the input to the SAR (from the adder out bus) with the selected data keys. Data keys 0 and 1 further define the comparison to either I/O or CPU mode (or both). This is a machine rather than program comparison and does not affect program execution time.

Master Chk Light

All machine check errors are grouped under a single indicator: master chk. Other locations of errors are on CPU roller 2 (position 6), common channel roller (positions 2, 4, and 5), and selector channel roller (position 3); LCS errors are indicated on CPU roller 1 (position 8).

Check Reset

Check reset is model-dependent in the Model 50. This switch is described under "Operator Intervention Controls."

Check Control

This switch varies in nomenclature throughout the System/360, and it is described under "Operator Intervention Controls." Similar functions of this switch in other models include: CPU check (Model 40), check control (Model 44), CPU check (Model 65), CPU check (Model 67), CPU chk (Model 75), machine check (Model 85), and CPU check stop (Model 91).

Concurrent Input/Output Capabilities

Each I/O device in operation places a load on its channel facilities. The magnitude of a load depends on a device's channel programming and its data transfer rate. In this manual, numeric factors are used to relate the loads caused by operation of I/O devices to the channel's abilities to sustain concurrent operation of the devices.

One or more numeric factors are specified for each I/O device and channel available with a Model 50. The numeric factors are presented in tables in the appendix to this manual and are used in arithmetic procedures for determining whether the operations of specific Model 50 input/output configurations are satisfactory.

Several procedures are provided for evaluating a configuration of I/O devices for concurrent operation on Model 50 channels. Use of the basic procedures will suffice for most configurations in determining whether operation is satisfactory; more detailed procedures are to be used only for configurations that appear to exceed Model 50 input/output capabilities.

Worst Case Loads

The evaluation factors and procedures allow for a worst case situation — when the most demanding devices in the configuration all make their heaviest demands on Model 50 I/O capabilities at the same time. Such a situation may not occur frequently, but it is the situation that the evaluation procedures place under test. If a particular configuration fails to pass testing, one or more devices may be expected to incur overrun or loss of performance.

Overrun

Overrun occurs when a channel does not accept or transfer data within required time limits during a read, read backward, or write operation. This data loss may occur when the total channel activity initiated by the program exceeds channel capabilities. Depending on the device, it may halt operation, or it may continue transferring data until the end of the block is reached.

An overrun causes a unit-check indication to be stored in the channel status word. An I/O interruption condition is generated at the conclusion of the operation. The cause for the unit check is indicated by turning on sense bit 5, the overrun bit, for subsequent interrogation.

Loss of Performance

Overrun occurs only on unbuffered I/O devices. Buffered devices are not subject to overrun. Instead, when buffer service is not provided within required time limits, the device merely waits for channel service. While it is waiting, the device is said to incur a loss of performance.

Conventions for Satisfactory Channel Programs

Execution of a channel program causes a load on channel and system I/O facilities. Some I/O devices require execution of a chain of commands, preparatory to transfer of a data block. However, the impact of the load caused by a channel program is not a simple function of the number of commands used: the sequence in which particular types of commands appear in a channel program is also a factor.

A type of command particularly significant to sequencing considerations is a control command that is executed at electronic speeds and that does not cause any mechanical motion. Such a command is executed as an immediate operation and provides device end in the initial status byte. When command chaining is specified in such an immediate operation, channel facilities are not disengaged from the channel program until such a chain ends or a command causing mechanical motion or data transfer is executed. Therefore, when immediate operations with device end in the initial status byte are chained together, fetching and execution of the ccw's may cause a heavy load on channel facilities. Such a load may cause excessive delay in channel service to one or more devices in the I/O configuration, with resultant overrun or loss of performance. For example, a chain of no-op commands can contribute heavily to channel loading. Such a programming convenience may cause a severe overrun situation for concurrently operating devices.

Data Chaining Considerations

A System/360 user is free to specify data chaining in channel programs whenever he chooses to do so, although a channel is able to transfer data at a faster rate, without overrun, when data chaining is not specified. The channel evaluation procedures and tables in this manual provide guidance in gauging the effects of data chaining operations.

The factors in Table 1 in the appendix allow for data chaining on the 2702, except for data chaining

on telegraph controls on a 2702, and as specified for other devices. The 2702 factors in Table 1 assume a count of 32 for data chaining. (The appendix is composed of Tables 1-7 and certain illustrations, which have been placed at the end of the manual so they can be removed easily for reference.)

To obtain maximum compatibility for data chaining channel programs, addressing resolution on double word boundaries and byte counts of 16 or greater are necessary.

Relationship of Conventions and Evaluation Procedures

The evaluation procedures are premised on channel programs having command sequences that provide efficient operation of I/O devices and avoid placing unnecessary loads on channel facilities. Channel programming conventions have been established to help I/O programmers avoid overrun situations.

Observance of channel programming conventions is fundamental to the selection of an I/O configuration that will permit concurrent operation of I/O devices in a satisfactory manner. The channel programming conventions described below are integral to the channel evaluation procedures. An evaluation yielding an indication of satisfactory channel performance is not dependable when channel programs written in violation of the conventions are used.

Scope of Conventions

1. The conventions relate to the sequence in which certain types of commands may be executed, not to their sequence in main storage.

2. The conventions define four classes of commands and the sequence in which they may be used.

3. The command sequences provided by the conventions are different for different types of devices. Sequences are provided for these devices:

DASD – 2302, 2303, 2311, 2314, 2321, 7320
Tape units – series 2400
Card units – 1442, 2501, 2520, 2540
Printers – 1403, 1443
Console – 1052
Communication adapters – 2701, 2702

4. The conventions relate to all the commands in a chain, including the CCW addressed by the CAW and the terminating CCW that does not specify any chaining.

5. The conventions do not relate to commands addressed by the CAW which do not specify any chaining.

6. The conventions relate only to the avoidance of overrun; they do not define invalid command sequences that are rejected by a channel, such as TIC to TIC, or that are rejected by a control unit. CCW sequences causing command reject are specified in the I/O device manuals.

Note that item 4 is of particular interest to I/O programmers working on segments of a single channel program; the rules still apply when one segment is chained to another segment.

The channel programming conventions in this manual are recommended to System/360 users, particularly in a multi-programming environment where a programmer is not aware of the overall load on channel facilities. Where a programmer controls or has knowledge of all I/O activity, he may establish somewhat less restrictive channel programming conventions of his own which may be particularly suited to his application and configuration.

Classes of Commands

The conventions establish four classes of commands. Commands that always cause mechanical motion are in one class. The other three classes encompass commands that are always executed at electronic speed plus commands that are sometimes executed at electronic speeds. An example of the latter is rewind, which is executed at electronic speeds when tape is already at load point. The three classes of commands having electronic-speed properties differ in the length of time required for their execution.

The conventions for the different devices specify classifications for the specific commands pertinent to each device.

The conventions define the four classifications by the sequence in which they may precede or follow other commands:

Class A Commands: These commands may be chained in any order, without restriction. Class A commands always cause mechanical motion.

Class B Commands: Only one Class B command may be chained between two Class A commands:

→ A → B → A. = permissible command chaining sequence
→ A → B → B → A. = command chaining sequence excluded by conventions

A Class B command may be substituted for a Class C or Class D command.

Class C Commands: A Class C command may appear only once in a channel program, and then only as the first command in a channel program; therefore, a Class C command may appear only in the location specified by the CAW:

CAW → C → A → B. = permissible program
CAW → A → C → A. = program excluded by conventions

A Class B command may be substituted for a Class C command:

CAW → B → A → B → A → B. = permissible program

Class D Commands: A Class D command may appear only as the last command in a channel program; it may not specify any chaining:

CAW → X → X → D. = permissible program
CAW → X → D → A. = program excluded by conventions

A Class B command may be substituted for a Class D command.

Some devices have conventions that exclude specific sequences of commands not excluded by the classifications above.

Some devices have conventions that allow a specific command sequence to be substituted for a single command of a specified class.

Command Classifications for I/O Devices

The rules below define classifications for specific commands used with a particular device. The bit pattern for each command code byte is specified to provide positive identification of commands.

Commands not classified may not specify any chaining and may be placed only in the location specified by the CAW. Each such command thus constitutes an entire channel program in which it is the only command. The sense command is used in this manner for all devices.

Direct Access Storage Devices: These command classifications are valid for all devices attached to a 2841 control unit.

Class A commands (any order):

Read	XXXX XX10	
Write, Search, Erase	XXXX XX01	
Space Record	0000 1111	
Calibrate	0001 0011	(Class A on 2311 only)
NoOp	0000 0011	(NoOp may be used only when preceded by a formatting write 0001 XX01 or 0000 0001)

Class B commands (not more than one between Class A commands):

TIC	XXXX	1000
SEEK	0000	0111
	000X	1011

These command chains have the properties of a single Class B command:

TIC → SEEK XXXX 1000 → 0000 0111 / 000X 1011

SEEK → TIC 0000 0111 / 000X 1011 → XXXX 1000

Class C commands (first CCW in program). These command chains have the properties of a single Class C command:

Seek → Set File Mask → TIC 0000 0111 / 000X 1011 → 0001 1111 → XXXX 1000

Class D commands (last CCW in program):

NoOp	0000	0011	(except when preceded by a formatting write)
Restore	0001	0111	(NoOp on other than 2311)

Excluded chains:

SEARCH → TIC → WRITE X011 0001 / X010 1001 → XXXX 1000 → 0000 X101

Data chaining may propagate through a TIC command for gap-only data-chaining, as described in the "Data Chaining in Gaps" part of this manual (included in the "channel characteristics" section).

Series 2400 Tape Units

Class A commands (any order):

Read	XXXX	XX10
Write	XXXX	XX01
Read backward	XXXX	1100
Forward space	0011	X111
Backspace	0010	X111
Write tape mark	0001	1111
Erase gap	0001	0111

Class B commands (not more than one between Class A commands):

TIC	XXXX	1000

Class C commands (first CCW in program):

Set Mode	XXXX	X011

This command chain has the properties of a single Class C command:

Set Mode → TIC XXXX X011 → XXXX 1000

Class D commands (last CCW in program):

Rewind	0000	0111
Rewind and Unload	0000	1111
NoOp	0000	0011

Mixed Mode Seven-Track Tape Operations: A routine may be used to select a tape unit, set its density mode, and then TIC to a desired channel program:

SIO → Set Mode / TIC } Class C sequence

The conventions require the CCW addressed by the TIC to be Class A.

If the tape applications involve mixed mode seven-track operations, the programmer may make provision for placing the proper set mode command in the location addressed by the CAW before SIO is issued, or the programmer may begin each channel program addressed by the TIC with an appropriate set mode command. Such an additional set mode command violates the convention for Class C commands and causes an additional load on channel facilities. Provision for the extra load is made in the multiplex mode evaluation procedure in this manual.

Card Units (1442, 2501, 2520, 2540):

Class A commands (any order):

READ	XXXX	XX10
WRITE	XXXX	XX01

Class B commands (not more than one between Class A commands):

TIC	XXXX	1000

Class C commands (first ccw in program):

CONTROL	XXXX	XX11

Class D commands (last ccw in program):

CONTROL	XXXX	XX11

Printers (1403, 1443):

Class A commands (any order):

WRITE	XXXX	XX01

Class B command (not more than one between Class A commands).

TIC	XXXX	1000

Class C commands (first ccw in program):

CONTROL	XXXX	XX11

Class D commands (last ccw in program):

CONTROL	0000	0011

Console (1052):

Class A commands (any order):

Read inquiry	0000	1010
Write auto carriage return	0000	1001
Write inhibit carriage return	0000	0001

Class B commands (not more than one between Class A commands):

TIC	XXXX	1000

Class C commands:

Not applicable

Class D commands (last ccw in program):

Control	XXXX	XX11

Communication Adapters (2701, 2702): Data chaining with or without TIC may be used for these adapters.

Class A commands (any order):

Write Dial Break Diagnostic write	XXXX	XX01
Read Prepare Inhibit Search Diagnostic read	XXXX	XX10

Class B commands:

Not applicable

Class C commands (first ccw in program):

Control*	XXXX	XX11

Class D commands (last ccw in program):

Control*	XXXX	XX11

*For a communication network of switch-type terminals, these two control commands are Class A:

Disable	0010	1111
Enable	0010	0111

Evaluating Heavily Loaded Channels

Concurrent operation of multiplex mode devices important to an application may be ensured by giving them higher priority than devices having less importance; higher priority may reduce the frequency of overrun or loss of performance incurred by a device.

When evaluating the performance of a system susceptible to channel overload conditions, consideration should be given to the relative ease of restarting an interrupted I/O operation. For example, an overrun on a communication line coupling two CPU's is handled more readily than a read overrun on a card read punch. Preferential priority may be given to devices that require manual intervention in response to an overrun condition.

Some circumstances may make it desirable to place devices with heavy load factors on the same selector channel, rather than on separate selector channels, in order to preclude interference with each other.

Evaluations should not ignore the characteristics of IBM Programming Systems packages:

1. Operating System/360
2. Tape Operating System
3. Disk Operating System
4. Basic Operating System
5. Basic Programming Support

These programs attempt to execute any start I/O instruction for which the channel and device are available. The programs that permit concurrent operation of more than one device on a multiplexer channel will not, however, initiate a burst mode operation on the multiplexer channel while any device subject to overrun is in operation on the multiplexer channel.

The impact on a selector channel of a load caused by operation of an unbuffered I/O device depends on the device's data transfer rate and on:

1. Whether data chaining is specified for the device.
2. Whether data chaining is specified for the device and transfer in channel (TIC) commands are used.
3. Whether data chaining is also specified for other selector channels.
4. Whether the data chaining is performed with a count of 16 or greater (long records).
5. Whether the initial byte of each chained record is located on a word boundary.

Overrun Test Exception

Under certain circumstances, the load on channel facilities caused by data chaining may be ignored in testing for channel overload. It then is not a consideration in selecting a load value or a selector channel load limit. The exception is valid only for direct access storage devices that are programmed in a certain way.

A channel program for direct access storage devices, such as an IBM 2311 Disk Storage Drive, must specify command chaining, and it may, of course, specify data-chaining operations. The time it takes a gap on a track to pass a read-write head on one of these devices is sufficient for the channel to perform a data-chaining operation. Gap time occurs in such operations as "write count, key, and data," where the gap time occurs between writing the count and the key, and between writing the key and the data.

If the program causes data chaining to occur only during gap time, the data-chaining load on channel facilities will not be additive to the device's data transfer load. Therefore, data chaining that occurs only during the gap time may be ignored in testing the channel for overrun.

Testing for Overrun

To determine whether selector channel operation can result in overrun, perform the following procedure:

1. Referring to Table 4 in the appendix, determine for each operating selector channel which device to be attached to it has the largest load value.
2. Determine which one of Tables 5-7 in the appendix is applicable. These tables are divided into three major groupings: load limit information pertinent (1) when both CPU and channel references (both data and CCW's) are made only to main storage, Table 5, (2) when CPU references may be made to the IBM 2361 Core Storage and channel references (both data and CCW's) are made only to main storage, Table 6, and (3) when both CPU and channel references (both data and CCW's) are made to the IBM 2361, Table 7. Each of the three tables is subdivided as follows.

No data chaining
Data chaining, multiplexer operating, LWN.
Data chaining, multiplexer operating, LWT.
Data chaining, multiplexer operating, LBN.
Data chaining, multiplexer operating, LBT.
Data chaining, multiplexer operating, SBN.
Data chaining, multiplexer operating, SBT.
Data chaining, multiplexer not operating, LWN.
Data chaining, multiplexer not operating, LWT.
Data chaining, multiplexer not operating, LBN.
Data chaining, multiplexer not operating, LBT.
Data chaining, multiplexer not operating, SBN.
Data chaining, multiplexer not operating, SBT.

LWN – long records, word boundaries, no TIC commands.
LWT – long records, word boundaries, with TIC commands.
LBN – long records, byte boundaries, no TIC commands.
LBT – long records, byte boundaries, with TIC commands.
SBN – short records, no TIC commands.
SBT – short records, with TIC commands.

Long records are those chained segments that have a length of 16 or more bytes. The indication of boundaries refers to the location of the first byte of the chained segment. "No TIC" or "with TIC" refers to the command immediately following any command that specifies data chaining. To qualify as "multiplexer operating," the multiplexer channel may be in burst mode or multiplex mode.

3. Enter the applicable table, determined in step 2, with the load values found in step 1. Find the line on which the appropriate configuration appears; then compare the load values from step 1 with the appropriate values listed on that line under "Maximum Load Values." (In the tables, "N/Op" means "not operating.") If any of the values from step 1 is greater than the appropriate value in the table, the configuration will overrun. If the values from step 1 are equal to or less than the appropriate values in the table and if no coefficients are listed for the overrun equation, the configuration will not overrun; if, however, coefficients are listed, perform step 4.

4. Substitute in the following equation the load values found in step 1:

$$PC_1 + QC_2 + RC_3 + MC_m \leq 100$$

where
- C_1 is the load value selected in step 1 for the first selector channel,
- C_2 is the load value selected in step 1 for the second selector channel,
- C_3 is the load value selected in step 1 for the third selector channel,
- C_m is the load value selected in step 1 for the multiplexer channel operating in burst mode, and
- $C_m = 0$ for the multiplexer channel operating in multiplex mode;

and where P, Q, R, and M are coefficients.

Then, determine whether the equation is satisfied for each set of coefficients given. If any of the applicable equations are not satisfied, the configurations will overrun.

Example 1: An IBM 2311 Disk Storage Drive Model 1 is on the first selector channel, and an IBM 2401 Magnetic Tape Unit Model 5 (1,600 bpi) is on each of the other two selector channels. References are to main storage only, the first selector channel is data chaining (long records, word boundaries, with TIC's), and the multiplexer channel is operating in multiplex mode.

Table 4 shows a load value for the 2311 Model 1 of .170 and a load value for the 2401 Model 5 of .152. The information just given shows that Table 5 is the applicable channel load-limit table. With a data-chaining configuration of X00, the maximum load value for the first selector channel is .269; for the second, .272; and for the third, .222. Although .170 is less than .269 and .152 less than .272 and .222, coefficients are listed in the table for X00, indicating that calculations must be made to determine whether the overrun equation is satisfied. Substituting, in the overrun equation, the load value for the devices and using the coefficients listed, the following calculations can be made:

$PC_1 + QC_2 + RC_3 \leq 100$
$(312 \times .170) + (82.7 \times .152) + (82.7 \times .152) \leq 100$
$53.0 + 12.6 + 12.6 \leq 100$
$78.2 \leq 100$

(Because the multiplexer channel is operating in the multiplex mode, $MC_m=0$.) The equation is satisfied, indicating that the configuration will not overrun.

Example 2: An IBM 2314 Direct Access Storage Facility Model 1 is on the first selector channel, an IBM 2314 Model 1 is on the second, and an IBM 2401 Magnetic Tape Unit Model 3 (800 bpi, with data conversion operating) is on the third, and an IBM 2401 Model 1 (800 bpi, with data conversion operating) is on the multiplexer channel. References are to main storage only, no selector channel is data chaining, and the multiplexer channel is operating in burst mode.

Table 4 shows a load value of .333 for the 2314 Model 1, a load value of .086 for the 2401 Model 3, and a load value of .030 for the 2401 Model 1. The preceding information shows that Table 5 is applicable. Note in Table 5 that when all three selector channels are operating and when the multiplexer channel is operating in burst mode, either of two lines of maximum load values may be used. Because of the load value of the 2314 Model 1, a check of the first line of maximum values indicates that the allowable load for the second selector channel would be exceeded. A check of the second line, however, shows that for the multiplexer channel, .030 is less than the maximum value (.044); for the first selector channel, .333 is equal to the maximum value (.333); for the second, .333 is equal to the maximum value; and for the third, .086 is less than the maximum value (.113). Because none of the maximum values is exceeded and because no overrun equation must be satisfied, the configuration will not overrun.

Example 3: An IBM 2314 Model 1 is data chaining (long records, word boundaries, no TIC's) on the first selector channel, a 2401 Magnetic Tape Unit Model 6 (1,600 bpi) is data chaining (long records, word boundaries, no TIC's) on the second selector channel, and the third selector channel is not operating. References are to main storage only, and the multiplexer channel is operating in multiplex mode.

Table 4 shows a load value for the 2314 Model 1 of .333 and a load value for the 2401 Model 6 of .222. The preceding information indicates that Table 5 is applicable. With a data-chaining configuration of X-X-N/Op, the maximum values are .352 for the first selector channel and .285 for the second. The load values of the devices do not exceed these limits, but coefficients are shown on the X-X-N/Op line, indicating that calculations must be made to determine whether the overrun equation is satisfied. Substituting, in the overrun equation, the load values for the devices and using the coefficients listed, the following calculations can be made:

$PC_1 + QC_2 \leq 100$
$(82.5 \times .333) + (350 \times .222) \leq 100$
$27.5 + 77.7 \leq 100$
$105.2 \leq 100$

(Because the third selector channel is not operating and the multiplexer channel is operating in multiplex mode, both RC_3 and $MC_m=0$.) The equation is not satisfied, indicating that the configuration will overrun.

Channel-to-Channel Adapter

The channel-to-channel adapter is unique in that it adjusts its speed to the varying demands of the other channels on the processing unit to which it is attached. The System/360 Model 50 operates at maximum efficiency when the channel-to-channel adapter is connected to the lowest-priority selector channel. If data chaining occurs on any other operating selector channel, it must be attached in the same manner; otherwise, the data chaining channel would be subject to overrun.

The formula for computing the rate at which data will be transferred over the channel-to-channel adapter is:

Rate (in megabytes) $= K - .75\,(C_A + C_B)$

where C_A and C_B are the load values (Table 4) for the devices operating on the higher priority (lower-numbered) selector channels (see following table) and K is shown for various conditions:

MULTIPLEXER CHANNEL	SELECTOR CHANNEL DATA CHAINING	K: MAIN STORAGE	K: 2361
Not operating	No chaining	.80	.37
Operating	No chaining	.60	.20
Operating	Long records	.30	.10
Operating	Short records	.05	.02

Note that the resultant calculated rate represents a worst-case condition for the system. When other channels are not transferring data (gap time, unoverlapped CPU time, etc.) the channel-to-channel connection will run at a faster rate. The instantaneous data rate may be .8 megabytes when all other channels are idle.

Also note that the other channel to which the adapter is attached imposes a limitation on the data transfer capability between systems. The data transfer rate cannot be higher than that permitted by the more limited system.

The effective channel-to-channel data transfer rate for a system, therefore, will vary between a maximum rate of .8 megabytes and a minimum rate as indicated by the preceding calculation or that imposed by the other system.

The multiplexer channel on the Model 50 can handle a burst mode I/O device with a load value not greater than .182. (See Table 4 for maximum capabilities under various conditions.) If multiplex mode devices are in operation when a burst mode operation is initiated on the multiplexer channel, they will overrun or lose performance when their ability to wait for channel service is exceeded.

Multiplex Mode Considerations

Concurrent operation of I/O devices on a multiplexer channel is governed by several variables, including the following:

1. Devices vary in their data transfer rates.
2. Devices have buffers varying in capacity from 1 byte to 132 bytes.
3. Devices vary in the number and type of CCW's needed for their operation.
4. Combinations of devices on the selector channels vary in the interference they cause.
5. The large number of I/O devices available for use on a multiplexer channel may be combined in many different ways.
6. Devices in a particular configuration may be physically connected in many different priority sequences.

The problem of determining whether a particular multiplexer channel configuration will run concurrently in a satisfactory manner has been reduced to arithmetic procedures in which a worksheet form and factors provided in tables in this manual are used.

Device Load

A numeric factor has been computed for each multiplex mode device to represent its load on channel facilities. It is called a device load. The factors are listed in Table 1 in the appendix.

Other factors are listed in Table 1 for use in considering the impact of higher priority devices on lower priority devices.

Device Wait Time

After a multiplex mode device requests channel service, it has a fixed length of time that it can wait for service. If the channel provides service within this length of time, the device operates satisfactorily. If, however, the channel does not service the device within the device's wait time, either of two things happens: if the device is not susceptible to overrun, it continues waiting; if it is, it loses data and subsequently causes an I/O interruption condition. For example, when an IBM 1403 Printer on an overloaded multiplexer channel fails to receive data within its particular wait time, it merely waits until service is provided by the multiplexer channel. The delay does not cause an interruption condition; nor is a new start I/O instruction required for selecting the 1403. The only effect is a lessening in performance. If an IBM 1442 Card Read Punch read operation does not receive data service within its wait time, however, overrun occurs.

Wait time factors for multiplex mode devices are listed in Table 1.

Device Priority on Multiplexer Channel

The priority of devices on a multiplexer channel is determined at the time of installation by the sequence in which they are connected to the channel. The cabling facilities provide considerable flexibility in the physical location and logical position of I/O devices.

Devices may have the priority sequence in which they are attached to the cable (select-out line priority) or the device most remote from the channel may be connected to have highest priority and the device nearest the channel connected to have lowest priority (select-in line priority).

Each device on the multiplexer channel cable may be connected (for selection) either to the select-out line, or to the select-in line. Thus, one or the other of the lines is specified in establishing priority for a desired physical layout of devices.

Priority assignments and machine room layout should be established during the physical planning phase of an installation so that cables for the I/O devices may be properly specified.

A major consideration in assigning priority to multiplex mode devices is their susceptibility to overrun. Devices are identified in this manual as being in one of three categories:

1. Devices subject to overrun, such as magnetic tape units.
2. Devices that require channel service to be in synchronization with their mechanical operations. For example, the IBM 2540 Card Read Punch has a fixed

mechanical cycle. Delay in channel service for such devices usually occasions additional delay due to synchronization lag.

3. Devices that do not require synchronized channel service. An IBM 2260 Display Station with its 2848 Display Control is such a device; it is entirely electronic in nature. An IBM 1443 Printer is another device that does not require synchronized channel service; it can begin printing as soon as its buffer is full and line spacing is completed. Any loss of performance by this category of devices is limited to that caused by channel delay in providing service.

Devices in the first category have a need for the highest priority. The devices in the last two categories may operate with reduced performance on an overloaded channel but are not subject to overrun; their control units have data buffers or an ability to wait for channel service. Devices in the second category, however, should have higher priority than those in the third category.

Within each category, devices are assigned decreasing priority in the order of their increasing wait time factors; smaller wait time factors should have higher priority. Wait time factors are listed in Table 1.

When devices that operate only in burst mode, such as magnetic tape or disk storage devices, are attached to the multiplexer channel, they should have lower priority than multiplex mode devices. Low priority devices take longer to respond to selection than do higher priority devices; a burst mode device need be selected only once for an operation, but a multiplex mode device must be selected for the transfer of each byte, or a short burst, of data.

The control unit determines whether a device operates on the multiplexer channel in burst mode or in multiplex mode.

Some devices, such as the IBM 2821 Control Unit, may operate on a multiplexer channel in either burst mode or in multiplex mode, as determined by the setting of a manual switch on the control unit's customer engineering panel. Such devices are assigned priority on the multiplexer channel as specified above.

A multiplexer channel can transfer data most rapidly in burst mode. Where an application uses only category 2 or 3 devices that have the mode choice, improved multiplexer channel efficiency may be obtained by operating the devices in burst mode.

Table 1 specifies whether a device operates in burst mode or in multiplex mode.

Interference from Priority Devices

The multiplexer channel sustains concurrent operations in multiplex mode by servicing one device at a time. The operating devices compete for service, and the multiplexer channel services them in the order of their priority.

Devices on the selector channels or higher priority devices on the multiplexer channel may force a lower priority mutiplex mode device to wait for channel service. The former are called priority devices, and the latter is called a waiting device.

When a priority device forces a waiting device to wait for channel service, the priority device is said to interfere with the lower priority device.

When there is more than one priority device, each of the priority devices may generate interference. All such interference must be considered in determining whether the waiting device will receive channel service before its wait time is exceeded.

The test procedures for concurrent operation of multiplex mode devices assume that a waiting device has made its request for channel service at the worst possible time, that is, when the priority devices will cause maximum interference during the waiting device's wait time.

The channel ordinarily works its way through the interference, and the waiting device is unaffected by the wait. If, however, heavy interference forces the waiting device to wait past its particular wait time, it will be subject to overrun or reduced performance, as determined by its type.

Priority Loads

To evaluate the effect of priority device interference on a waiting device, a numeric priority load is computed.

Three factors are considered in determining a priority load:

1. The control load caused by execution of CCW's, including chaining and transfer in channel operations.
2. The priority device's data transfer load.
3. The wait time of the device being evaluated.

Note that since a priority load is a function of wait time, a *fixed* priority load cannot be established for a priority device; the priority load caused by a priority device must be computed as a function of a particular waiting device's wait time.

Ranges of Wait Times

The relationship between the interference generated by a priority device (expressed as "priority load") and various wait times is shown in Figure 10. The abscissa relates to device wait times. The short wait time shown results in a heavy priority load; the longer wait time falls in a part of the curve showing much less priority load. This curve shows that the impact of a

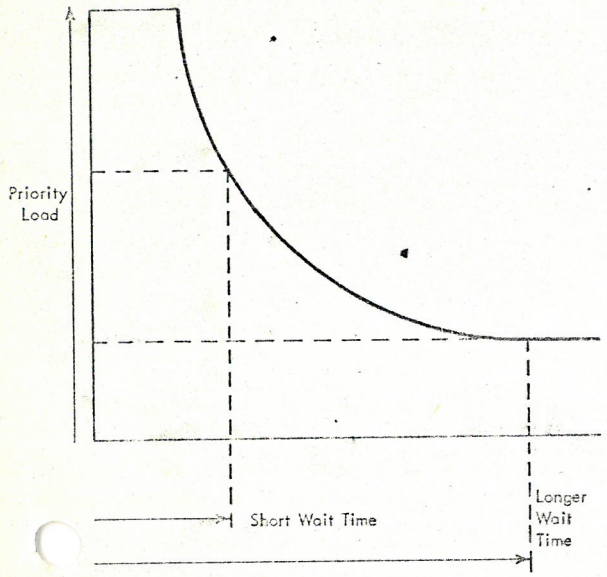


Figure 10. Priority Load Curve

priority device on a waiting device is more intense for a waiting device with a short wait time than it is for a device with a long wait time.

Two factors, called A and B, are provided in this manual which relate each device's priority load curve to different wait times. The priority load curve was considered in segments related to different time intervals, and an A and a B factor were computed for each curve segment. These factors are used to compute the priority load for a waiting device having a wait time that falls within the range of the interval established for the curve segment.

Multiple A and B Factors: Table 1 lists the A and B factors for each Model 50 input/output device.

Some devices have only one set of A and B factors. Other devices have more than one set. Each set has an associated priority time factor that represents the beginning of the time interval over which the A and B factors are effective.

Priority Time Factors: The priority time factors in Table 1 are used in the evaluation procedure to identify the A and B factors to be used.

As each waiting device is evaluated on a multiplexer channel worksheet, its wait time is used to select a set of A and B factors for each priority device.

Each set of A and B factors in Table 1 has a priority time factor next to it that specifies the beginning of a time interval significant to that set of A and B factors. The range extends from the priority time factor specified for that set to the priority time factor specified for the next set, if any. The end of the last interval is assumed to be unbounded. For example, a device may have three sets of A and B values which describe the priority load function over three contiguous intervals. Figure 11 shows the priority time factors and A and B factors as they appear in Table 1 for an IBM 1443 Printer Model N1 with a 13-character set.

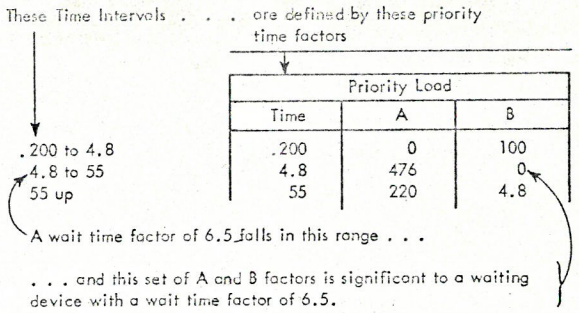


Figure 11. Use of Priority Time Factors

Priority Load Formula

The A and B factors and wait time factors in Table 1 have been provided for use in a formula that yields the priority load which occurs when a priority device interferes with a waiting device.

The sum of the B factor and the quotient obtained by dividing the A factor by the wait time factor of the waiting device is the priority load. The arithmetic looks like this:

A/wait time + B = priority load

Table 1 provides priority load factors for data chaining byte counts of 20 and 100; the factors may be interpolated or extrapolated for other counts by using a linear function of 1/count.

The procedure for using arbitrary counts is:

1. Use the wait time to select A and B factors for a count of 20, compute the priority load, and call it L_{20}:

$$\frac{A_{20}}{\text{wait time}} + B_{20} = L_{20}$$

2. Repeat step 1 for a count of 100, and call the result L_{100}:

$$\frac{A_{100}}{\text{wait time}} + B_{100} = L_{100}$$

3. Compute the priority load for the desired count:

$$L_{100} + \left(\frac{25}{\text{count}} - .25\right)\left(L_{20} - L_{100}\right) = \text{Priority load for count}$$

Previous Load

A waiting multiplex mode device may be forced to wait for channel facilities, not only by devices with higher priority, but also by a device with lower priority that is in operation when the waiting device requests

channel facilities. This interference is called a "previous load" and must be added to the priority load caused by priority devices. The device with lowest priority on the channel has no previous load; a zero value is used in the addition. Previous load factors are provided in Table 1.

Load Sum

Several load factors relating to multiplex mode operations have been described:

Device load (contributed by waiting device)
Priority load (contributed by each priority device)
Previous load (contributed by a lower priority device)

These loads are added together to form a load sum for each waiting device. The load sum represents the total load on system channel facilities under a worst case condition when:

1. All priority devices are causing maximum priority loads.
2. Any lower priority device, already in operation, is making maximum demands on channel facilities (previous load).
3. The waiting device places its maximum device load on channel facilities.

A step-by-step procedure for computing load sums is given in "Multiplex Mode Evaluation Procedure."

Multiplex Mode Channel Load Limit

A numeric factor of 100 has been established as the multiplex mode channel load limit. If a load sum exceeds 100, loss of performance or overrun is indicated during worst case situations.

Lost Performance Time

A loss of performance indicated by a load sum greater than 100 is caused by the waiting device's having been forced to wait past its wait time. This reduced performance may be inconsequential in many cases, as shown in the following illustration. The total length of time the device waits for channel service during a worst case situation is computed as:

$$\frac{\text{load sum x wait time (ms)}}{\text{multiplex mode channel load limit of 100}} = \text{total delay in channel service (in ms)}$$

Wait time is subtracted from the quotient to find the amount of time lost:

total delay in ms − wait time = lost performance time in ms

By relating the amount of time lost to the device's normal operating cycle time, the effect on performance may be seen:

$$\frac{\text{lost time}}{\text{cycle time}} \times 100 = \text{percentage loss of performance}$$

For a hypothetical device having:

Wait time = 20 ms
Load sum = 120
Cycle time = 200 ms

The arithmetic is:

$$\frac{\text{120 load sum x 20 wait time}}{\text{100 (limit)}} = \text{24 ms total delay}$$

and,

24 ms total delay − 20 ms wait time = 4 ms lost performance time

and

$$\frac{\text{4 lost time}}{\text{200 cycle time}} \times 100 = \text{2 percent loss of performance (occurring only during worst case situations)}$$

Multiplex Mode Evaluation Procedure

The following step-by-step procedure is used with a *System/360 Multiplexer Channel Worksheet*, Form X24-3407, shown in Figure 12. (Figures 12-15 and 17-21 are part of the appendix; they are at the end of the manual so they can be removed easily for reference.)

Most of the steps call for an entry to be made on the worksheet. Each circled number shown on the worksheet in Figure 13 refers to the numbered step in the following procedure. For example, a circled number 1 is shown at the top of the worksheet in each of the two spaces that receive the entries called for by step 1. As an additional aid in seeing where entries are made on a worksheet, refer to Figure 14, which shows a worksheet that has been completed for a configuration specified in the next part of this section.

The following procedure assumes that the selector channel configuration has already been defined and evaluated (see "Selector Channel Loading").

1. Enter the system identification and the date.
2. Identify for each operating selector channel the device that has the greatest load value.
3. For the devices entered in step 2, enter the "time, A, B" sets listed under "Selector Channel Priority Load" in Table 1.
4. Arrange the multiplex mode devices proposed for simultaneous operation in sequence by priority categories (1, 2, and 3), as specified for the devices in the Table 1 column headed "Key." Within the priority categories, arrange the devices according to increasing wait time; the device with the smallest wait time appears first (receives highest priority). Then enter the devices on the worksheet in the sequence just established.
5. For the first device entered in step 4, enter from Table 1 the wait time; the "time, A, B" sets listed under

"Multiplexer Channel Priority Load"; the previous load; and the device load.

6. Repeat step 5 for each remaining device entered in step 4; the lowest priority device has no previous load factor.

7. Compare the wait time factor of the first waiting device being evaluated to the time factors of the priority device for the first selector channel; enter the set of A and B factors that relate to the time interval that includes the wait time (that is, the set that is on the same line with the largest time factor that is less than the wait time factor).

8. Repeat step 7 with the time factors of the priority devices listed for the other selector channels in step 2.

9. For the second and each other waiting device on the multiplexer channel, compare its wait time to the time factors given for each of the selector channel priority devices (steps 7 and 8) and also for each of the multiplexer channel devices with higher priority; enter the appropriate set of A and B factors (that is, the set that is on the same line with the largest time factor that is less than the wait time).

10. For each multiplexer channel waiting device, add the selected A factors and enter the A sum.

11. Divide the A sum by the wait time factor for the device and enter the quotient.

12. For each multiplexer channel waiting device, add the B values, the quotient found in step 11, the device load, and the previous load, and enter this sum as the load sum. The load sum must be less than or equal to 100 for satisfactory operation of the waiting device. If a 2702 Transmission Control has a load sum greater than 100, further evaluation is required; consult the "Load Sums for 2702" part of this section.

Worksheet Example

The following I/O configuration is evaluated for use on a Model 50 (Figure 14):

Selector channel 1	2311 Disk Storage Model 1, no data chaining
Selector channel 2	2401 Magnetic Tape Unit Model 6 (180 kb/sec), no data chaining
Selector channel 3	2401 Magnetic Tape Unit Model 6 (180 kb/sec), no data chaining
Multiplexer channel	2501 Card Reader Model B2 (EBCD)
	2821 Control Unit
	2540 Card Read Punch Model 1 (reading EBCD)
	1403 Printer Model N1 (1,100 lines a minute)
	1403 Printer Model N1 (1,100 lines a minute)
	1052 Printer-Keyboard Model 7 (2150 Console)

The completed multiplexer channel worksheet for the given configuration is shown in Figure 14; it shows satisfactory operation for all multiplex mode devices: no load sum exceeds 100.

Worksheet Entries for 2821

Each device attached to an IBM 2821 Control Unit is evaluated as if it were a separate control unit. Each device has its own channel service requirements and is evaluated in a separate column on the multiplexer channel worksheet. The space provided on the worksheet may not be sufficient for evaluating all the pertinent devices, and the user must then make appropriate adjustments.

The priority sequence for 2821 devices is:

IBM 2540 Card Read Punch Model 1 (Reading)
IBM 2540 Card Read Punch Model 1 (Punching)
Printers
  Printer Control No. 1
  Printer Control No. 2
  Printer Control No. 3

IBM 2702 Considerations

The IBM 2702 Transmission Control may connect a variety of communication terminals to a multiplexer channel; 1-15 or 1-31 terminal lines may be connected.

The 2702 uses delay lines for storage of data and control information. The information circulates in the delay lines and may be accessed for transfer to or from the multiplexer channel or to or from a terminal.

When priority devices force a 2702 to wait for channel service, additional delay may occur in the 2702 because of time required for synchronization with the delay line. Such additional delay exists only for the 2702 and does not affect other devices on the multiplexer channel.

A bit of information takes a certain length of time to go once around a delay line. A 2702 with a capacity for 15 terminal lines takes 0.480 millisecond per revolution, and a 31-line 2702 has a delay line revolution time of 0.992 millisecond. The number of communication lines attached to a 2702 has a direct bearing on how long it can wait for channel service. Maximum wait time exists when only one communication line is used. Each additional line in operation reduces the time a 2702 can wait for channel service.

In addition, the data transfer speed of a terminal affects 2702 wait time; a high-speed line cannot wait for channel service as long as a lower-speed line. Therefore, the wait time factors specified in Table 1 vary with the type of terminal control and number of lines available. The factors shown in Table 1 for the 2702 are for all lines operating at the same speed.

Worksheet Example With Two 2702's and a 2821

The following Model 50 i/o confiiguration is evaluated:

Selector channel 1	2303 Drum Storage Model 1, no data chaining
Selector channel 2	2311 Disk Storage Model 1, no data chaining
Selector channel 3	2401 Magnetic Tape Unit Model 5 (120 kb/sec), no data chaining
Multiplexer channel	2702 Transmission Control – 15 1030's (Terminal Control II) at 600 bps
	2702 Transmission Control – 31 1050's (Terminal Control I) at 135.5 bps
	2821 Control Unit
	2540 Card Read Punch Model 1 (Reading EBCD)
	2540 Card Read Punch Model 1 (Punching EBCD)
	1403 Printer Model N1 (1,100 lines a minute)
	1403 Printer Model N1 (1,100 lines a minute)
	1052 Printer-Keyboard Model 7

The completed worksheet for this configuration is shown in Figure 15.

The load sum for the IBM 1403 Printer Model N1 with fifth priority is 102.5, and the load sum for the 1403-N1 with sixth priority is 114.3; these sums indicate a loss in performance for both printers under worst case conditions.

The maximum length of time that channel service to the 1403-N1 with sixth priority would be delayed may be computed:

$$\frac{\text{waiting device load sum} \times \text{waiting device wait time}}{100} = \text{maximum delay in ms}$$

Values for the 1403-N1 are:

114.3 x 15.7/100 = 17.9 ms

The 1403-N1 can wait 15.7 ms for its buffer to be serviced; in this worst case situation, it must wait an additional 2.2 ms (17.9 – 15.7 = 2.2). The 1403-N1 ordinarily prints a line in 54.5 ms. An increase to 56.7 ms during a period of maximum priority loads is an increase of 4 percent.

When a 2702 has a load sum in excess of 100, the following "Load Sums for 2702" is pertinent.

When a 2702 contributes a priority load to any device having a load sum in excess of 100, the "Priority Load Factors for 2702" is pertinent.

Load Sums for 2702: The wait time, device load, and previous load factors specified in Table 1 are conservative values, and in most instances their use in computing 2702 load sums gives an indication of satisfactory operation. However, when a 2702 load sum exceeds 100, a more detailed examination is called for.

To this end, a special analysis procedure unique to the 2702 is provided in the next section of this manual. The procedure uses a special 2702 worksheet for analysis, and resolution is to a single delay line cycle.

When the special analysis indicates satisfactory operation of the 2702, attention may be returned to the multiplexer channel worksheet for evaluation of the next waiting device. If, however, the special analysis load sum still indicates an overrun, some of the communication lines may have to be connected to another 2702 in order to eliminate overrun.

In a system with a large number of terminal lines, construction of a probabilistic model may lead to the conclusion that the frequency of overrun will not be great enough to be objectionable for a particular application.

Special Analysis of 2702 Performance

If the multiplexer mode evaluation procedure finds a load sum greater than 100 for an IBM 2702 Transmission Control, the more sophisticated performance analysis given here may indicate that operation is satisfactory.

The special analysis assumes that all attached communication lines will request service during a single delay line revolution and that a scanning sequence will occur that gives service last to the highest speed communication line. The analysis reveals whether, considering priority loads, the number of delay line revolutions available is sufficient for the total delay line revolution requirements of all communication lines.

It is seldom necessary to test every communication line's requirements for delay line revolutions. After a communication line tests satisfactorily, a projection is made of both the minimum and the maximum number of revolutions needed to service the remaining communication lines. When a projected maximum is fewer than the maximum revolutions needed for the highest speed remaining line, satisfactory operation is indicated and no further analysis is required. Similarly, if a projected minimum is greater than the maximum revolutions needed for the highest speed remaining line, overrun is indicated, and the analysis is complete.

The projections are made on the 2702 worksheet as the procedure progresses. Figure 16 illustrates the relationship of the two projections to the maximum number of revolutions needed for the highest speed line. Satisfactory operation is indicated in Figure 16 whenever an upper curve crosses the line indicating the maximum number of revolutions before overrun.

To determine the number of delay line cycles required by a particular communication line, tables of factor values are provided in this manual for use with the *2702 Worksheet*, Form X24-3406.

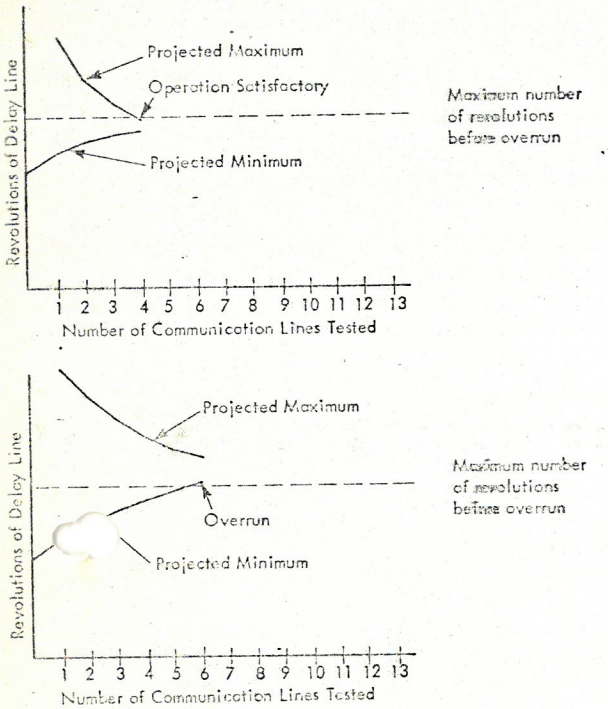


Figure 16. Projection of Delay Line Revolution Requirements

The factors are used to compute a load sum occurring during the servicing of each communication line. The load sum consists of priority loads caused by selector channel priority devices and by multiplexer channel priority devices, plus a device load factor and a previous load factor for the terminal being tested. The various factors are entered on the 2702 worksheet and are used to compute a load sum which is then compared to the load limit already specified on the worksheet for that particular delay line revolution (first, second, etc).

If the load sum is greater than the specified load limit, the communication line under consideration requires an additional delay line revolution. The projected minimum time for service is increased one revolution and tested. If overrun is not indicated, the next column of the 2702 worksheet is used to compute a new load sum which is compared, etc.

If, however, the first load sum mentioned in the previous paragraph was not greater than the specified load limit, adequate service is indicated for the communication line under consideration, and if it was serviced in one revolution, or if it is the last communication line to be considered, satisfactory operation of the 2702 is indicated. But if the communication line serviced was not the last one and was not serviced in a single revolution, it is necessary to see if the remaining communication lines can be serviced within the number of revolutions remaining to them. A new projection of the maximum time for service is made.

In this analysis, no communication line remaining to be evaluated will take more revolutions than the communication line for which satisfactory service was just indicated; therefore, if the number of revolutions it required is multiplied by the number of remaining lines and the results compared to the remaining number of revolutions available, a low or equal comparison indicates satisfactory operation, and the analysis is complete. A high comparison indicates a need to test the next communication line. This is done by transferring values on the worksheet in use to a new 2702 worksheet and testing the next communication line for satisfactory operation. A load sum is computed and compared with the load limit. Comparison results have the significance already described.

Special Analysis Procedure

The following procedure is used with a 2702 worksheet, shown in Figure 17. (Figures 17-21 are part of the appendix; they are at the end of the manual so they can be removed easily for reference.) Most of the steps call for an entry to be made on the worksheet. The circled numbers shown on the worksheet in Figure 18 refer to the numbered steps in the procedure. The procedure is shown in flowchart form in Figure 19. This flowchart also is keyed to the numbered steps in the procedure.

To use the 2702 worksheet, perform the following:

1. In three places, enter the number of communication lines proposed for attachment.
2. Enter 1 as the line number.
3. Subtract the step 2 entry from the step 1 entry and enter the remainder as K.
4. In Table 2, find the smallest Nmax value that is listed for the terminals proposed for attachment and enter that value.
5. From Table 2, enter the device load and previous load values shown for the terminal selected in step 4.
6. Identify the selector channel and multiplexer channel priority devices in the leftmost column in the order of their priority (copy from the multiplexer channel worksheet).
7. For t1, enter a value of zero.
8. Enter zero values in the A1 and B1 spaces.
9. Enter the t2 value.

When j = 1, the t2 value is:
t1 + .464 for a 15-line 2702
t1 + .976 for a 31-line 2702
When j > 1, the t2 value is:
the previous t2 + .480 for a 15-line 2702
the previous t2 + .992 for a 31-line 2702

10. Use the t2 value just entered to select A2 and B2 factors from the left-hand column of the multiplexer channel worksheet. Copy the selected A's and B's from that worksheet into the A2 and B2 spaces on the 2702 worksheet.

NOTE: If the 2702 that is being evaluated has a configuration of terminal lines that operate at different speeds, use the A and B factors in Table 1 for the highest speed line or compute the factors by using the procedure described in the "Priority Load Factors for 2702" part of this section.

11. Enter the A1 sum and B1 sum; this step is performed only once per worksheet. (When the line number is 1, the A1 sum and the B1 sum are zeros.)

12. Multiply the B1 sum by the t1 value and enter the product; this step is performed only once per worksheet.

13. Determine and enter the A2 sum and the B2 sum.

14. Multiply the B2 sum by the t2 value and enter the product.

15. Subtract the A1 sum from the A2 sum and enter the A remainder.

16. Subtract the B1 *product* from the B2 *product* and enter the B remainder.

17. Add the A remainder, the B remainder, the device load, and the previous load, and enter the total as the load sum.

18. Compare the load sum to the appropriate load limit shown at the bottom of the worksheet. If the load sum is not greater, adequate communication line service is indicated; go to step 21. If it is greater, the communication line needs another delay line revolution; go to step 19.

19. Add 1 to the last-entered n value; enter that sum in the next n space.

20. If the new n value is greater than the Nmax value, no additional delay line revolution is available for the communication line. Overrun is indicated. If the new n value is not greater than Nmax, go to step 9 and repeat the performance analysis for the new delay line revolution. When j is greater than 8, the analysis must be continued on another worksheet; each new load limit is then computed by adding a load limit increment to the old load limit.

Load limit increment:

48.0 for a 15-line 2702
99.2 for a 31-line 2702

21. The step 18 load sum was not greater than the load limit, indicating adequate channel service for the communication line under test. If this line is the last communication line (that is, the line number = the number of lines), or if it was serviced by the first delay line revolution (j = 1), satisfactory operation of the 2702 is indicated. But if the communication line serviced was not the last one and was not serviced by the first revolution (that is, if j is greater than 1), add its n value to the product of its revolution number minus one (j − 1) times K, and enter the sum in the space at the bottom of the worksheet, where n + (j − 1) K is printed.

22. If the value just entered is not greater than the Nmax, satisfactory operation of the 2702 is indicated. If it is greater, get a new 2702 worksheet.

23. Make the same "number of lines" entry for the new worksheet as was on the old worksheet. Enter the n value last used on the old worksheet in the spaces under t1 and j = 1.

24. Enter a new line number one greater than the old line number.

25. Subtract the step 24 entry from the step 23 entry for "number of lines" and enter the remainder as K.

26. Make the same entries made in steps 4, 5, and 6.

27. Enter a new t1 value by adding 0.048 to the old t2.

28. Use the new t1 value to select priority device A and B factors from the multiplexer channel worksheet and enter the factors in the A1 and B1 spaces.

29. Go to step 8.

2702 Special Analysis Example

Figure 20 shows a multiplex mode evaluation of the following I/O configuration:

Selector channel 1	2314 Direct Access Facility Model 1, no data chaining
Selector channel 2	2311 Disk Storage Model 1, no data chaining
Selector channel 3	2311 Disk Storage Model 1, no data chaining
Multiplexer channel	2702 Transmission Control – 15 1030's (Terminal Control II) at 600 bps
	2702 Transmission Control – 15 1030's (Terminal Control II) at 600 bps
	2702 Transmission Control – 15 1030's (Terminal Control II) at 600 bps
	2702 Transmission Control – 5 1030's (Terminal Control II) at 600 bps and 10 1050's (Terminal Control I) at 135.5 bps
	1052 Printer Keyboard Model 7

Figure 20 shows an excessive load sum of 108.3 for the 2702 having fourth priority; use of the 2702 special analysis procedure is indicated.

Figure 21 shows the completed 2702 Worksheet used in the special analysis. The 2702 special analysis procedure in the preceding part of this section was used in making the entries.

Figure 21 (Sheet 1) shows service to be completed for the 1030 on the eighth cycle; the projected maxi-

mum number of cycles needed is 120, which is greater than the Nmax of 30; a new 2702 Worksheet is used.

Figure 21 (Sheets 2-4) continue the analysis. The Nmax factor of 30 does not change; the procedure assumes that if all terminals request service, the 1030 may be the last to be serviced. The Nmax factor for the highest speed line is always used.

Figure 21 (Sheet 4) shows a satisfactory load sum on the 28th delay line revolution. This indication of satisfactory operation for the 2702 completes the evaluation of the communication configuration for concurrent operation.

Priority Load Factors for 2702

An IBM 2702 Transmission Control may have terminal lines attached that all operate at the same speed. When this is the case, the priority load A and B factors listed in Table 1 (for the type of terminal control and the number of lines attached) are used for the multiplex mode evaluation.

An IBM 2702 may have a configuration of terminal lines that operate at different speeds. Where this is the case, the priority load factors in Table 1 for the highest speed line may be used; the A and B factors used are those listed for the number of lines attached. When these factors are used, the slower speed lines receive undue weight; but if their use does not cause any load sum to exceed 100, satisfactory operation is indicated, and the disparity in line speeds may be ignored.

If their use indicates unsatisfactory multiplexer channel operation, however, a more accurate assessment of the situation may be made:

1. Retain the first set of "time, A, B" factors already entered on the multiplexer channel worksheet for the priority 2702 and also retain the time factor from the second set.

2. As described below, compute a new second set of "A, B" factors, which then replace the second set already entered.

New load sums are then computed. Any new load sum that is less than or equal to 100 indicates satisfactory operation for the load sum device.

Each new second set of "A, B" factors is computed as specified in steps 1-3. An example computation is shown immediately following step 5.

1. Select from Table 2 a *b* factor for each type of terminal. Multiply each selected *b* factor by the number of terminal lines having that *b* factor, and add all the products. The sum of the products is the *new B factor.*

2. Subtract the new B factor from the B factor specified in the first set of "A, B" factors already entered; then multiply the remainder by the time factor retained from the second set.

3. Add the A factor specified in the first set of "A, B" factors already entered to the product found in step 2. The sum is the *new A factor.*

4. Substitute the "A, B" factors just determined in place of the second set of "A, B" factors previously entered on the multiplexer channel worksheet for the priority 2702.

5. Repeat steps 1-5 for any remaining 2702 priority devices and then consider the new "A, B" factors in computing new load sums for the devices previously found to have excessive load sums.

A new second set of "A, B" factors can be computed for a 2702 with a mix of line speeds as shown in the following example:

Consider a 15-line 2702 to which is attached:
One 1030 line (Terminal Control II) —at 600 bps
Ten 1050 lines (Terminal Control I) at 135.5 bps

Step 1. From Table 2.
1030: 1 x .1150 = .1150
1050: 10 x .0235 = .2350 +
new B = .3500

Step 2. From first set: B = 7.60
From step 1: new B = .35 −
7.25
From second set: t = 7.21 x
57.27

Step 3. From first set:
From step 2: 57.27 +
60.86

Step 4. Previous priority load factors (from Table 1):

TIME	A	B
.200	3.59	7.60
7.21	46.4	1.67

New priority load factors:

TIME	A	B
.200	3.59	7.60
7.21	60.86	.35

IBM 2703 Considerations

For this evaluation, see Tables 1.1 and 1.2. Table 1.1 covers the channel evaluation factors; Table 1.2 tabulates 2703 priorities.

In the calculations:

Base A has N(A) lines with P(A) priority and t_e (A) wait time.

Base B has N(B) lines with P(B) priority and t_e(B) wait time.

Base C has N(C) lines with P(C) priority and t_e(C) wait time.

Priority (P) is obtained from Table 1.2.

The wait time of the highest speed line is t_e; it is obtained from Table 1.1.

T_e(X) is the effective wait time of base A, B, or C (denoted by X).

$T_e(X) = \frac{t_c(X)}{N_e(X)}$ where N_e is the number of lines on all bases which may obtain service before the last line on this base (X).

$N_e(X) = N(X) + S(X, Y_1) + S(X, Y_2)$ where:

$$S(X, Y) = \left[\frac{N(X)/P(X)}{N(Y)/P(Y)}\right] N(Y) = \left[\frac{r(X)}{r(Y)}\right] N(Y)$$
$$= [R(X, Y)]\, N(Y), \text{ and } R(X, Y) \text{ max} = 2.$$

The smallest T_e determines the critical base.
The largest N_e determines the priority load.

Procedure for one 2703:

1. Find most critical base (smallest T_e).
2. Compute device load, $L_d = \frac{3.86\, N_e}{t_c - K}$

   ($K = 0.015\, N_e$ except when another 2703 or a buffered device is operating simultaneously on the multiplexer channel; in that case, $K = 0$.)
3. Multiplexer channel priority load L_h is found by using "Multiplex Mode Evaluation Procedure" for all higher priority multiplexer channel devices. Selector channel priority load L_s is found by a similar procedure. Table 1 is entered with $T = t_c$ (wait time of the highest speed line of the critical base).
4. Previous load $L_p = \Sigma L_h$ of all lower priority devices up to L_p max $= L_d$.
5. $L_d + L_p + \Sigma L_h + \Sigma L_s \leq 100$. Load sums in excess of 100 indicate an overrun on the 2703.

Procedure for two 2703's:

1. Same as preceding Step 1 for each 2703.
2. Same as preceding Step 2 for each 2703.
3. (In this procedure, Steps 3 and 4 for one 2703 are combined.) The effect of priority load and previous load of one 2703 on another is determined by considering the fact that a 2703 always delays a short time between its own consecutive requests for service, thus allowing other devices to utilize the channel.

In the following equations:

(*Code 1*) = All devices of a priority higher than the higher priority 2703.
(*Code 2*) = All devices of a priority lower than the lower priority 2703.
(*Code 3*) = The other 2703.

The first of the two 2703's is the more critical; it limits the allowable Selector Channel Load.

For the higher priority 2703:

$L_p = L_p$ (*Code 3*) $+ L_p$ (*Code 2*)

$$L_p = \frac{3.86\, N_e \text{ max } (\textit{of second 2703})}{t_c \quad (\textit{of first 2703})} + \Sigma L_h \text{ (Code 2)}$$

up to L_p max $= L_d$ (*of first 2703*).

$L_h = \Sigma L_h$ (*Code 1*)

For the lower priority 2703:

$L_h = L_h$ (*Code 3*) $+ L_h$ (*Code 1*)

$$L_h \text{ (Code 3)} = \frac{3.86\, N_e \text{ max } (\textit{of first 2703})}{t_c \quad (\textit{of second 2703})}$$

up to L_h max $= L_d$ (*of second 2703*).

L_h (*Code 1*) is found as in Step 3 of the procedure given for two 2703's.

$L_p = L_p$ (*Code 2*)

$L_p + L_h$ (*Code 3*) $\leq L_d$ (*of second 2703*).

When more than two 2703's are attached, they may be evaluated as successive pairs. The procedure for the first pair is the same as that for two 2703's. The third 2703 does not obtain service until one of the first pair has completed service of N_e max lines. The remaining lines of the other 2703 then alternate with the third 2703. The last 2703 in the series is loaded by all pairs of 2703's having higher priority.

$$L_h = \frac{3.86\, N_t}{t_c}$$

where N_t is the total number of higher priority lines that may be serviced before the last 2703 is able to complete its service requirements.

Example 1 (One 2703)

Base A: $N(A) = 88$ lines of 1050
$P(A) = 6$, and $t_c(A) = 59.5$
Base B: $N(B) = 24$ lines of 1030
$P(B) = 4$, and $t_c(B) = 13.3$
Base C: $N(C) = 24$ lines @ 2400 bps
$P(C) = 8$, and $t_c(C) = 6$

$$\frac{N(A)}{P(A)} = \frac{88}{6} = 14.7 = r(A)$$

$$\frac{N(B)}{P(B)} = \frac{24}{4} = 6 = r(B)$$

$$\frac{N(C)}{P(C)} = \frac{24}{8} = 3 = r(C)$$

$$R(X, Y) = \frac{r(X)}{r(Y)}$$

$$\begin{aligned} N_e(A) &= N(A) + S(A, B) + S(A, C) \\ &= N(A) + R(A, B)\, N(B) + R(A, C)\, N(C) \\ &= 88 + \frac{14.7}{6}(24) + \frac{14.7}{3}(24) \quad (\textit{See Note}) \\ &= 88 + 2(24) + 2(24) \\ &= 88 + 48 + 48 \\ &= 184 \end{aligned}$$

$$T_e(A) = \frac{t_c(A)}{N_e(A)} = \frac{59.5}{184} = 0.323$$

Note: $\frac{14.7}{6} > 2, \therefore R(A, B) = 2$

$\frac{14.7}{3} > 2, \therefore R(A, C) = 2$

$$\begin{aligned} N_e(B) &= N(B) + S(B, C) + S(B, A) \\ &= N(B) + R(B, C)\,N(C) + R(B, A)\,N(A) \\ &= 24 + \frac{6}{3}(24) + \frac{6}{14.7}(88) \\ &= 24 + (2)(24) + (0.408)(88) \\ &= 24 + 48 + 36 \\ &= 108 \end{aligned}$$

$$\begin{aligned} T_e(B) &= \frac{t_c(B)}{N(B)} \\ &= \frac{13.3}{108} = 0.123 \end{aligned}$$

$$\begin{aligned} N_e(C) &= N(C) + S(C, A) + S(C, B) \\ &= N(C) + R(C, A)\,N(A) + R(C, B)\,N(B) \\ &= 24 + \frac{3}{14.7}(88) + \frac{3}{6}(24) \\ &= 24 + (0.204)(88) + (0.5)(24) \\ &= 24 + 18 + 12 \\ &= 54 \end{aligned}$$

$$\begin{aligned} T_e(C) &= \frac{t_c(C)}{N_e(C)} \\ &= \frac{6}{54} = 0.111 \end{aligned}$$

Base C is critical base, N_e max = 184

Device Load = $L_d = \frac{3.86\,N_e}{t_c - K} = \frac{3.86\,(54)}{6 - 0} = 35$

Assume $L_h = 0$

Assume that a 1403 Model 3 has lower priority.

Then:

$L_p = L_h$ of 1403 @ t = 6

(From Table 1: A = 185, B = 0)

$$L_p = \frac{185}{6} = 31$$

$L_d + L_p + \Sigma L_h + \Sigma L_s \leq 100$

$L_d + L_p + L_h = 66, \Sigma L_s + 66 \leq 100, \Sigma L_s \leq 34$

L_s cannot exceed 34 (one channel of 2314 or two channels of 2311's).

Example 2 (Two 2703's)

Add a second 2703 to the configuration in Example 1. Compute the first 2703 as before.

For the second 2703:

Base A: N(A) = 32 lines of 1050
P(A) = 2, and $t_c(A) = 59.5$

Base B: N(B) = 24 lines of 1030
P(B) = 4, and $t_c(B) = 13.3$

Base C: N(C) = 16 lines @ 2400 bps, 6-bit code autopolling
P(C) = 8, and $t_c(C) = 4.1$

$\frac{N(A)}{P(A)} = \frac{32}{2} = 16 = r(A)$

$\frac{N(B)}{P(B)} = \frac{24}{4} = 6 = r(B)$

$\frac{N(C)}{P(C)} = \frac{16}{8} = 2 = r(C)$

$R(X, Y) = \frac{r(X)}{r(Y)}$

$$\begin{aligned} N_e(A) &= N(A) + S(A, B) + S(A, C) \\ &= N(A) + R(A, B)\,N(B) + R(A, C)\,N(C) \\ &= 32 + \frac{16}{6}(24) + \frac{16}{2}(16) \qquad (\textit{See Note}) \\ &= 32 + 2\,(24) + 2\,(16) \\ &= 32 + 48 + 32 \\ &= 112 \end{aligned}$$

Note: $\frac{16}{6} > 2, \therefore R(A, B) = 2$

$\frac{16}{2} > 2, \therefore R(A, C) = 2$

$\frac{6}{2} > 2, \therefore R(B, A) = 2$

$$\begin{aligned} T_e(A) &= \frac{t_c(A)}{N_e(A)} \\ &= \frac{59.5}{112} \\ &= 0.53 \end{aligned}$$

$$\begin{aligned} N_e(B) &= N(B) + S(B, C) + S(B, A) \\ &= N(B) + R(B, C)\,N(C) + R(B, A)\,N(A) \\ &= 24 + \frac{6}{16}(32) + \frac{6}{2}(16) \qquad (\textit{See Note}) \\ &= 24 + 0.375\,(32) + 2\,(16) \\ &= 24 + 12 + 32 \\ &= 68 \end{aligned}$$

Note: $\frac{16}{6} > 2, \therefore R(A, B) = 2$

$\frac{16}{2} > 2, \therefore R(A, C) = 2$

$\frac{6}{2} > 2, \therefore R(B, A) = 2$

$$\begin{aligned} T_e(B) &= \frac{t_c(B)}{N_e(B)} \\ &= \frac{13.3}{68} \\ &= 0.196 \end{aligned}$$

$$N_e(C) = N(C) + S(C, A) + S(C, B)$$
$$= N(C) + R(C, A)\,N(A) + R(C, B)\,N(B)$$
$$= 16 + \frac{2}{16}(32) + \frac{2}{6}(24)$$
$$= 16 + 0.125\,(32) + 0.333\,(24)$$
$$= 16 + 4 + 8$$
$$= 28$$

$$T_e(C) = \frac{t_c(C)}{N_e(C)}$$
$$= \frac{4.1}{28}$$
$$= 0.146$$

Base C is critical base, N_e max = 112

Device Load (*of second 2703*) = L_d (*of second 2703*)
$$= \frac{3.86\,(28)}{4.1} = 26.5$$

L_d (*of first 2703*) = 35 (*from Example 1*)
L_h (*Code 1*) = 0 (*from Example 1*)

Assume a 1403 Model 3 has lower priority.

For the first 2703:
$L_h = \Sigma L_h$ (*Code 1*) = 0
$L_p = L_p$ (*Code 3*) + L_p (*Code 2*)
$$L_p = \frac{3.86\,N_e\,\text{max}\,(\textit{of second 2703})}{t_c\quad(\textit{of first 2703})} + L_p\,(\textit{Code 2})$$
$$= \frac{3.86\,(112)}{6} + \frac{185}{6} = 72 + 31 = 103$$

But:
L_p max = L_d (*of first 2703*) = 35
$\therefore L_p = 35$
$L_d + L_p + L_h = 70$, $\Sigma L_s + 70 \leq 100$, $\Sigma L_s \leq 30$
L_s cannot exceed 30

For the second 2703:
$L_h = L_h$ (*Code 3*) + L_h (*Code 1*)
But:
L_h (*Code 1*) = 0
$$L_h\,(\textit{Code 3}) = \frac{3.86\,N_e\,\text{max}\,(\textit{of first 2703})}{t_c\quad(\textit{of second 2703})}$$
$$= \frac{3.86\,(184)}{4.1} = 175$$

But:
L_h max = L_d (*of second 2703*) = 26.5
$$L_p = L_p\,(\textit{Code 2}) = \frac{185}{4.1} = 45$$

$L_p + L_h$ (*Code 3*) $\leq L_d$ (*of second 2703*)
But:
$45 + 175 > 26.5$
$\therefore L_p = 0$
$L_d + L_p + L_h = 53$

Synchronization Tendency of Buffer Servicing

When evaluation of a multiplex mode configuration shows loss of performance for several buffered devices, additional analysis may show a reduction of the indicated loss because of the tendency of multiple buffered devices to synchronize, to a greater or lesser extent, their use of channel facilities.

By estimating the delays involved in servicing the device's buffers and by relating the delays to the devices requests for channel service, it may be discovered that some of the buffered priority devices do not interfere with buffered waiting devices to the extent premised in the evaluation procedure. The procedure assumes a random relationship between the operations of the various I/O devices that may not apply to buffered devices.

For example, if both of two card readers in operation request channel service at the same time, the higher priority device will force the other device to wait; and having once waited, the second card reader will next request channel service after the first device has already made its next request for channel service. The two new requests will not coincide unless the first card reader has been similarly delayed by some other device. This synchronization effect tends to organize buffered devices' requests for channel service into a sequence that enables the channel to service them on a rotating basis, and a loss of performance premised on random channel service requests may be significantly reduced.

The analysis of the synchronization effect is done by laying out the operating cycles of the buffered devices in their priority sequence, one below another, on a millisecond scale. The devices that operate satisfactorily are drawn with a zero starting point. A new starting point is established on the millisecond scale for each device found to incur delay. The resulting synchronization pattern may be studied to see which buffered device priority loads may be ignored in computing new load sums.

Operation cycle times are specified in Systems Reference Library manuals for the devices.

A channel operation on the Model 50 interferes with CPU use of main storage whenever the channel requests access to main storage. Additional CPU interference is generated because the channels use some CPU facilities.

The amount of CPU interference caused by an I/O device over a period of time depends on its data transfer rate and its channel programming. Table 3 lists the factors used to compute Model 50 channel interference with the CPU.

When an application requires the concurrent operation of I/O devices, it must first be determined that the devices will operate without overrun. This is done as described in the channel loading sections of this manual.

Channel Interference Procedure

After an indication of satisfactory operation has been found, channel interference with the CPU may be computed, after which the available CPU time may be determined. The procedure has three steps:

1. Examine record lengths, data transfer rates, gap times, device operating cycle times, etc., and establish an I/O operation time span pertinent to the application.
2. Calculate the CPU interference caused during the I/O time span by:
   a. Execution of start I/O's
   b. Execution of CCW's
   c. Data transfer
   d. I/O interruptions
3. Subtract the sum of total CPU interference time in milliseconds from the I/O time span. The difference is the milliseconds of time available for CPU operations during the time span.

Dividing the available CPU time by the time span and multiplying by 100 gives the percentage of available CPU time for the application considered:

$$\frac{\text{available CPU time}}{\text{time span}} \times 100 = \% \text{ available CPU time}$$

Available CPU Time Example

Application

Tape-to-printer operation.

Configuration

The Model 50 will use an IBM 2403 Magnetic Tape Unit and Control Model 2 (800 bytes per inch, data conversion feature not in operation) on the first selector channel, and an IBM 1403 Printer Model N1 (1,100 lines a minute) on the multiplexer channel.

Analysis

The tape read operation will handle 1,000-byte tape blocks data chained into 10 scattered 100-byte blocks in main storage which lie on word boundaries.

The printer may be programmed with a start I/O for each line of print, or it may be programmed with one start I/O and nine chained commands for each 10 lines of print. The difference in CPU interference caused by the two approaches is examined in the example.

Evaluation of Concurrent Operation

The 2403-2 load value of .076 is less than the selector channel load limit of .400 (for references to main storage only, a data-chaining configuration of X-N/Op-N/Op, with multiplexer operating, and LWN); it will run satisfactorily. Only one set of "A, B" factors are listed for the priority 2403-2 (800 bpi, data conversion not in operation, and data chaining with a count of 100); they are A = 0 and B = 7.3. The 1403 has a wait time of 15.7.

The formula A/wait time + B = priority load yields:

$$\frac{0}{15.7} + 7.3 = 7.3 \text{ priority load}$$

The 1403 device load is 11.8 and its previous load is 0.6. A load sum is computed:

Priority load	=	7.3
Device load	=	11.8
Previous load	=	0.6
Load sum	=	19.7

The load sum is less than 100; satisfactory concurrent operation is indicated.

Arithmetic for Channel Interference Example

The computation of available CPU time demonstrated below uses the following three-step procedure:

1. Establish the I/O time span.
2. Compute the channel interference with CPU.
3. Subtract the sum of interference from the time span to find the available CPU time.

The information necessary to execute step 2 is found in Tables 1 and 3.

Table 1 provides data transfer rates, gap times, and cycle times for I/O devices.

Table 3 provides the microseconds of CPU delay per byte transferred, both on the selector channels and the multiplexer channel, plus the microseconds of CPU delay per CCW execution and end interruption.

Step 1: Establish time span.

The time needed to read this 1,000-byte tape record block (24.7 ms) is listed on the tape timing card, *IBM System/360 Magnetic Tape Record Characteristics for IBM 2400 Series Magnetic Tape Units*, Form X22-6837, or can be computed by using the formula on the same card:

Model 1 – ms per record block = 16.0 + 0.0333N
Model 2 – ms per record block = 8.0 + 0.0167N
Model 3 – ms per record block = 5.3 + 0.0111N
N = Number of bytes in record block

The time to print 10 lines is 10 times the 1403-N1 print cycle time (listed in Table 1):

10 x 54.5 = 545 milliseconds to print 10 lines

Because the tape and printer operations will be overlapped, the longer printer time of 545 ms is the time span pertinent to the application configuration.

Step 2: Compute channel interference with CPU.

Tape transfer interference time is the product of the number of bytes in the tape block multiplied by the selector chanel byte transfer CPU interference factor, .95 microsecond (from Table 3):

1000 x .95 = 950 microseconds tape transfer interference

Tape data chaining interference time is the product of the number of data chaining operations per record block and the selector channel data chaining CPU interference factor, 7 microseconds (from Table 3):

9 x 7 = 63 microseconds tape data chaining interference

Printer transfer interference is found by computing the product of the number of characters per print line times the multiplexer channel byte transfer CPU interference factor, 10 microseconds (from Table 3), times the number of print lines handled during the time span:

100 x 10 x 10 = 10,000 microseconds printer transfer interference

Printer command time is the product of the number of chained commands per time span times the multiplexer channel command chaining CPU interference factor, 29 microseconds (from Table 3):

9 x 29 = 261 microseconds printer command interference

Start I/O interference factors are referenced from the instruction timing section of this manual, end interruption factors are refereuced from Table 3, and total interference time may be computed:

Tape	MICROSECONDS
Start I/O	35
Transfer interference (as previously computed)	950
Data chaining interference (as previously computed)	63
Channel end interruption	22
Printer	
Start I/O	50
Transfer interference	10,000
Command chaining interference	261
Channel end interruption	28
Device end interruption	58
Total interference time =	11,467

Step 3: Compute available CPU time in milliseconds.

Available CPU time is found by subtracting the interference time from the time span:

545 – 11.5 = 533.5 milliseconds available CPU time

Step 4: Compute available CPU time as a percentage.

The CPU interference may be expressed as a percentage by dividing the interference time by the time span and multiplying by 100:

(11.5/545) x 100 = 2.1 percent CPU interference

Command Chaining Efficiency

By ignoring printer data transfer interference, attention may be focused on the CPU interference caused by channel control functions for the printer. In the example, command chaining may be eliminated if a start I/O is used for each print line. A comparison of the CPU interferences caused by the two methods reveals which is more efficient, as shown below.

Printer control interference using command chaining is:

	MICROSECONDS
Start I/O	50
Command chaining	261
Channel end	28
Device end	58
Printer control interference using command chaining =	397

If command chaining is not used for printer operation each print line occasions a start I/O and two end interruptions, and the following arithmetic applies:

	MICROSECONDS
Start I/O – 50 x 10	500
Device end – 58 x 10	580
Channel end – 28 x 10	280
Total	1,360

Printer control interference using no command chaining is 1,360 microseconds.

The use of a start I/O for each print-line occasions 1,360 – 397 = 963 microseconds of additional CPU interference. Command chaining is clearly more efficient.

The instruction time tables presented here are divided into two groups:

Group 1 This group of instruction times provides the average time for all instructions used with the Model 50.

Group 2 This group of instruction times contains the detailed timing formulas for all variable field length (VFL) instructions used with the Model 50.

All symbols used in the formulas should be interpreted in accordance with the Legend for System/360 Timing.

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.

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 an asterisk in the instruction name column), an additional 0.5 microsecond 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. (Exception: supervisor call. See item 4 under "Timing Assumptions.") Interruption times are stated separately in a table following instruction times.
6. All timings provided include both decoding and execution of the instruction.

Timing Assumptions

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

1. For decimal add (AP) and subtract (SP) instructions, the length of the first operand (i.e., destination field) is assumed to be equal to or greater than the length of the second operand (i.e., source field).
2. In the TRANSLATE and TEST (TRT) instruction, it is assumed that a nonzero byte from a translate and test table is found.
3. The instruction times for the floating-point instructions depend on the number of hexadecimal digits that are preshifted and postshifted, as well as the number of times recomplementation of the result occurs. Each of the floating-point instruction times listed is an *average* of actual execution times. Although each value is the most accurate that can be given, the actual time is data-dependent.
4. The supervisor call (SVC) instruction includes interruption time.
5. Start I/O, Halt I/O and Test I/O instruction times and device end interruption times include average control unit response times to interface signals from the channels.

Average Timing Formulas

INSTRUCTION	FORMAT	MNEMONIC	INSTRUCTIONS & DATA IN PROCESSOR STORAGE
Add	RR	AR	$3.25 + G_2$
Add*	RX	A	$4.00 + G_2$
Add Decimal	SS	AP	$16.22 + 1.13N_1 + 0.53N_2 + T_1\,(3.00 + 1.00N_1)$
Add Halfword*	RX	AH	$5.50 + G_2$

INSTRUCTION	FORMAT	MNEMONIC	INSTRUCTIONS & DATA IN PROCESSOR STORAGE
Add Logical	RR	ALR	3.25
Add Logical*	RX	AL	4.00
Add Normalized (Long)	RR	ADR	9.81
Add Normalized (Long)*	RX	AD	11.94
Add Normalized (Short)	RR	AER	7.97
Add Normalized (Short)*	RX	AE	8.73
Add Unnormalized (Long)	RR	AWR	9.57
Add Unnormalized (Long)*	RX	AW	11.70
Add Unnormalized (Short)	RR	AUR	7.48
Add Unnormalized (Short)*	RX	AU	8.24
AND	RR	NR	5.00
AND*	RX	N	5.75
AND	SI	NI	6.50
AND	SS	NC	$12.19 + 1.50N$
Branch and Link	RR	BALR	$3.00 + 1.25F_2$
Branch and Link*	RX	BAL	4.50
Branch on Condition	RR	BCR	$2.75 + F_1$
Branch on Condition*	RX	BC	$3.00 + F_1$
Branch on Count	RR	BCTR	$3.25 + F_1 + 0.50F_2$
Branch on Count*	RX	BCT	$3.50 + F_1$
Branch on Index High	RS	BXH	$4.50 + F_1$
Branch on Index Low or Equal	RS	BXLE	$4.50 + F_1$
Compare	RR	CR	3.25
Compare*	RX	C	4.00
Compare Decimal	SS	CP	$12.03 + 1.63N_1 + 0.47N_2$
Compare Halfword*	RX	CH	5.50
Compare Logical	RR	CLR	3.00
Compare Logical*	RX	CL	4.00
Compare Logical	SI	CLI	4.50
Compare Logical	SS	CLC	$10.00 + 1.00B$
Compare (Long)	RR	CDR	8.08

INSTRUCTION	FORMAT	MNEMONIC	INSTRUCTIONS & DATA IN PROCESSOR STORAGE
Compare (Long)*	RX	CD	9.00
Compare (Short)	RR	CER	6.75
Compare (Short)*	RX	CE	7.29
Convert To Binary*	RX	CVB	C_1:10.75 + 2.00H C_2:17.50 + 2.00H C_3:23.75 + 2.00H
Convert To Decimal*	RX	CVD	C_1:13.00 + 2.00H C_2:23.25 + 2.00H C_3:28.75 + 2.00H
Divid	RR	DR	28.13
Divide	RX	D	28.88
Divide Decimal	SS	DP	$15.00 + 16.00N_1$ $- 13.13N_2 + T_6$ $[13.00(N_1 - N_2) - 6.00]$
Divide (Long)	RR	DDR	72.25
Divide (Long)*	RX	DD	75.00
Divide (Short)	RR	DER	21.75
Divide (Short)*	RX	DE	22.50
Edit	SS	ED	$9.63 + 2.38N_1$ $+ 0.50N_2 + 0.50N_5$
Edit and Mark	SS	EDMK	$10.13 + 2.38N_1$ $+ 0.50N_2 + 0.50N_5$
Exclusive OR	RR	XR	5.00
Exclusive OR*	RX	X	5.75
Exclusive OR	SI	XI	6.50
Exclusive OR	SS	XC	$12.19 + 1.50N$
Execute*	RX	EX	$E_1 = 4.75 + E$ $E_2 = 5.00 + E$ $E_3 = 6.50 + E$ $E_4 = 6.75 + E$
Halt I/O	SI	HIO	$B_1 = 31$ $B_2 = 94$ $B_4 = 8$
Halve (Long)	RR	HDR	$7 + 1.5H_2 + 0.5L_1$
Halve (Short)	RR	HER	$6.5 + H_2 + 0.5L_1$
Insert Character*	RX	IC	5.00

INSTRUCTION	FORMAT	MNEMONIC	INSTRUCTIONS & DATA IN PROCESSOR STORAGE
Insert Storage Key	RR	ISK	5.25
Load	RR	LR	2.50
Load*	RX	L	4.00
Load Address*	RX	LA	2.75
Load and Test	RR	LTR	2.50
Load and Test (Long)	RR	LTDR	4.25
Load and Test (Short)	RR	LTER	3.75
Load Complement	RR	LCR	2.75
Load Complement (Long)	RR	LCDR	4.25
Load Complement (Short)	RR	LCER	3.75
Load Halfword*	RX	LH	4.75
Load (Long)	RR	LDR	3.50
Load (Long)*	RX	LD	6.00
Load Multiple	RS	LM	3.00 + 2.00GR
Load Negative	RR	LNR	2.88
Load Negative (Long)	RR	LNDR	3.75
Load Negative (Short)	RR	LNER	3.25
Load Positive	RR	LPR	3.00
Load Positive (Long)	RR	LPDR	3.75
Load Positive (Short)	RR	LPER	3.25
Load PSW	SI	LPSW	7.50
Load (Short)	RR	LER	2.75
Load (Short)*	RX	LE	4.00
Move	SI	MVI	4.50
Move	SS	MVC	10.69 + 1.00N
Move Numerics	SS	MVN	12.19 + 1.50N
Move With Offset	SS	MVO	11.88 + 1.00N_1 + 0.88N_2
Move Zones	SS	MVZ	12.19 + 1.50N
Multiply	RR	MR	26.13 – 1.50K_1
Multiply*	RX	M	27.38 – 1.50K_1

INSTRUCTION	FORMAT	MNEMONIC	INSTRUCTIONS & DATA IN PROCESSOR STORAGE
Multiply Decimal	SS	MP	$12.84 + 11.40N_1 - 9.03N_2 + T_6 [6.50(N_1 - N_2) + 0.75]$
Multiply Halfword*	RX	MH	$22.50 - 1.50K_1$
Multiply (Long)	RR	MDR	47.00
Multiply (Long)*	RX	MD	49.75
Multiply (Short)	RR	MER	20.75
Multiply (Short)*	RX	ME	21.50
OR	RR	OR	5.00
OR*	RX	O	5.75
OR	SI	OI	6.50
OR	SS	OC	$12.19 + 1.50N$
Pack	SS	PACK	$10.38 + 1.13N_1 + 0.75N_2$
Read Direct	SI	RDD	$6.50 + ED$
Set Program Mask	RR	SPM	2.75
Set Storage Key	RR	SSK	6.75
Set System Mask	SI	SSM	5.50
Shift Left Double	RS	SLDA	$5.00 + q_4 + r_4 + 2.00S_6 + 0.50 [R_4(1 - S_6) + S_5]$
Shift Left Double-Logical	RS	SLDL	$4.00 + q_4 + r_4 + 0.50R_4$
Shift Left Single	RS	SLA	$4.00 + 0.50 [q_4 + r_4 + S_7 + R_4]$
Shift Left Single-Logical	RS	SLL	$4.00 + 0.50 [q_4 + r_4 + R_4]$
Shift Right Double	RS	SRDA	$4.00 + q_4 + r_4 + 0.50[Q_4 + R_4 + S_5]$
Shift Right Double-Logical	RS	SRDL	$4.00 + q_4 + r_4 + 0.50R_4$
Shift Right Single	RS	SRA	$4.00 + 0.50 [q_4 + r_4 + R_4]$
Shift Right Single-Logical	RS	SRL	$4.00 + 0.50 [q_4 + r_4 + R_4]$
Start I/O	SI	SIO	B_1:50 B_2:50 B_4:35

INSTRUCTION	FORMAT	MNEMONIC	INSTRUCTIONS & DATA IN PROCESSOR STORAGE
Store*	RX	ST	4.00
Store Character*	RX	STC	4.50
Store Halfword*	RX	STH	5.00
Store (Long)*	RX	STD	6.00
Store Multiple	RS	STM	$3.00 + 2.00GR$
Store (Short)*	RX	STE	4.00
Subtract	RR	SR	$3.25 + G_2$
Subtract*	RX	S	$4.00 + G_2$
Subtract Decimal	SS	SP	$16.22 + 1.13N_1 + 0.53N_2 + T_1[3.00 + 1.00N_1]$
Subtract Halfword*	RX	SH	$5.50 + G_2$
Subtract Logical	RR	SLR	3.25
Subtract Logical*	RX	SL	4.00
Subtract Normalized (Long)	RR	SDR	11.34
Subtract Normalized (Long)*	RX	SD	13.60
Subtract Normalized (Short)	RR	SER	8.75
Subtract Normalized (Short)*	RX	SE	9.54
Subtract Unnormalized (Long)	RR	SWR	10.80
Subtract Unnormalized (Long)*	RX	SW	12.85
Subtract Unnormalized (Short)	RR	SUR	8.25
Subtract Unnormalized (Short)*	RX	SU	9.04
Supervisor Call	RR	SVC	12.75
Test and Set	SI	TS	4.50
Test Channel	SI	TCH	B_1:6.0 B_2:83.5 B_3:6.5 B_4:6.0
Test I/O	SI	TIO	D_1:12.0 D_2:83.5 D_3:6.0 D_4:38.0 D_5:49.0 D_6:6.0 D_7:29.0 D_8:18.5

INSTRUCTION	FORMAT	MNEMONIC	INSTRUCTION & DATA IN PROCESSOR STORAGE
Test Under Mask	SI	TM	$5.50 - 0.50G_4$
Translate	SS	TR	$6.00 + 4.50N$
Translate and Test	SS	TRT	$7.75 + 3.00B$
Unpack	SS	UNPK	$9.88 + 1.00N_1 + 0.63N_2$
Write Direct	SI	WRD	7.00
Zero and Add	SS	ZAP	$13.22 + 1.13N_1 + 0.41N_2$

Variable Field Length Instructions

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. 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 (for Model 50 this is four bytes). Thus, the number of word boundary crossovers is one less than the number of words spanned by the field.

Add Decimal — AP

$$17.88 + 4.5NWBL_1 + 2.13NWBL_2 + T_1\,[4.00 - 2.00T_2 + 4.00\,(1 - T_2)\,NWBL_1] + T_{16}\,[1.5 + 0.5\,(N_2 - N_1)] + 1.00G_2$$

And — NC

$$13.69 + 4.00NWBL_1 + 2.00NWBL_2$$

Compare Decimal — CP

$$13.63 + 4.50NWBL_1 + 1.88NWBL_2 + 0.50N_1 + 0.5ABV + T_{16}\,[0.5\,(N_2 - N_1) + 2.50\,(NWBL_2 - NWBL_1] + T_{13}\,[0.50\,(NWBL_1 - NWBL_2)]$$

Compare Logical — CLC

$$11.00 + 2.00NWBB_1 + 2.00NWBB_2 - 1.25T_{14}$$

NOTE: The compare logical operation is terminated when an unequal condition is found. All bytes of the first operand word that contain the unequal condition are processed with the corresponding bytes of the second operand.

Divide Decimal — DP

$$15.00 + 2.50NWBL_1 + 2.00NWBL_2 + 2.00NWBQ_1 + 1.5NWBR_1 + 15.00N_1 - 13.50N_2 + T_6\,[13.00\,(N_1 - N_2) - 6.00]$$

NOTE: This divide decimal formula assumes that the quotient digits have random value. If the quotient contains zeros, the following times can be subtracted:

If $N_2 \leq 4$, subtract 4.00 μs/zero digit
If $N_2 > 4$, subtract 8.00 μs/zero digit

Edit — ED

$$11.00 + 3.50NWBL_1 + 2.00NWBL_2 + 1.50N_1 + 0.50N_5$$

Edit and Mark — EDMK

$$11.50 + 3.50NWBL_1 + 2.00NWBL_2 + 1.50N_1 + 0.50N_5$$

Exclusive OR — XC

$$13.69 + 4.00NWBL_1 + 2.00NWBL_2$$

Move Characters — MVC

$$11.69 + 2.00NWBL_1 + 2.00NWBL_2$$

Move Numerics — MVN

$$13.69 + 4.00NWBL_1 + 2.00NWBL_2$$

Move With Offset — MVO

$$12.00 + 1.00NWBL_1 + 2.50NWBL_2 + 1.00N_1 + T_{13}\,[1.00\,(NWBL_1 - NWBL_2) - 0.50\,(N_1 - N_2)]$$

Move Zones — MVZ

$$13.69 + 4.00NWBL_1 + 2.00NWBL_2$$

Multiply Decimal — MP

$$12.80 + 4.00NWBL_1 + 2.0NWBL_2 - 0.50NWBL_1L_2 + 10.40N_1 - 9.40N_2 + T_6\,[0.75 + 6.50\,(N_1 - N_2)] + 1.50MQ_1$$

NOTE: This multiply decimal formula assumes that multiplier digits have random values. If the multiplier contains zeros, the following times can be subtracted:

If $N_2 \leq 4$, subtract 2.7 μs/zero digit
If $N_2 > 4$, subtract 5.45 μs/zero digit

OR — OC

$13.69 + 4.00NWBL_1 + 2.00NWBL_2$

Pack — PACK

$10.75 + 1.5NWBL_1 + 2.5NWBL_2 + 1.0N_1$
$+ T_{11}\,[0.63 - 0.5N_1 + 0.25N_2]$

Subtract Decimal — SP

$17.88 + 4.50NWBL_1 + 2.13NWBL_2$
$+ T_1\,[4.00 - 2.00T_2 + 4.00\,(1 - T_2)\,NWBL_1]$
$+ T_{16}\,[1.5 + 0.5\,(N_2 - N_1)] + 1.00G_2$

Translate — TR

$6.50 + 2.0NWBL_1 + 4.0N$

Translate and Test — TRT

$8.25 + 2.0NWBL_1 + 2.5N - 1.75T_9$

Unpack — UNPK

$11.00 + 2.00NWBL_1 + 2.50NWBL_2 + 0.50N_1$
$+ T_3\,[0.13N_1 - 0.25N_2 + 0.5]$

Zero and Add Decimal — ZAP

$13.88 + 2.00NWBL_1 + 2.13NWBL_2$
$+ 0.50N_1 + 2.00T_1T_2$
$+ T_{16}\,[0.50\,(N_2 - N_1)$
$+ 0.50\,(NWBL_2 - NWBL_1) + 1.50]$

Large Capacity Storage Timing

Instruction times are lengthened when instructions and/or data are located in the 2361 Core Storage, which has an 8.0-microsecond cycle time. The following section contains timing factors which adjust the times previously quoted for instructions and data located in 2.0-microsecond Processor Storage. Various combinations of instruction and data location, both with and without 2361 interleaving, are included.

The timing adjustments are presented in groups, according to the number of references to data in storage (none, one, or two). Within these groups, many instructions may be affected in the same way, and these are classified together. There are three classes for RR instructions (no reference to data in storage), four classes for RX, RS, and SI instructions (one reference to data in storage), and five classes for SS instructions (two references to data in storage).

In the instructions following, the times are given in microseconds. The symbol *w/o* indicates without interleaving, *w* indicates with interleaving. Where an asterisk (*) appears, it signifies no change with interleaving.

RR Instructions

**Class 1:* Fixed- and floating-point multiply and divide instructions in LCS, add 3.0.

Class 2: Successful Branch.
*instruction in LCS, branch to main, add 3.0.
instruction in LCS, branch to LCS, total time:
w/o 12.0
w 10.5
*instruction in main, branch to LCS, add 4.0.

Class 3: All other instructions in LCS, add 3.0, minimum total:
w/o 8.0
w 7.25

RX, RS, and SI Instructions

Class 1: Fixed- and floating-point multiply and divide.
*instruction in LCS, data in main, add 3.0.
instruction and data in LCS, add:
w/o 8.5
w 7.0
(for long floating point)
w/o 10.5
w 9.0
*instruction in main, data in LCS, add 3.0 (except add 4.5 for long floating point).

Class 2: Successful Branch.
*instruction in LCS, branch to main, add 3.0.
instruction in LCS, branch to LCS, total time:
w/o 12.0
w 10.5
*instruction in main, branch to LCS, add 4.0.

Class 3: Shift instructions in LCS, add 3.0, minimum total:
w/o 8.0
w 7.0

Class 4: All others.
instruction in LCS, data in main, add 3.0, minimum total:
w/o 8.0
w 7.0
instruction and data in LCS,
all double floating point, NI, OI, XI, total:
w/o 24.0
w 22.5
radix conversion, add:
w/o 14.0
w 12.5
load and store multiple, total:
w/o 8.0 + 8.0GR
w 8.0 + 6.5GR

others, total:
w/o 16.0
w 14.5
instruction in main, successive data in LCS,
*all double floating point, NI, OI, XI, 16.0 total.
*radix conversion, add 10.0.
load and store multiple, total:
w/o 0.5 + 8.0GR
w 0.5 + 6.5GR
others, add 3.0, minimum total:
w/o 8.0
w 7.0
instruction in main, scattered data in LCS,
*all double floating point, NI, OI, XI, add 8.0.
*radix conversion, add 9.0.
load and store multiple, total:
w/o 8.0GR
w 6.5GR
*others, add 3.0.

SS Instructions

Class 1: Decimal Multiply (MP).
instruction in LCS, data in main, add:
w/o 9.0
w 7.5
instruction and data in LCS, add:
w/o $22.0 + 9.5NWBL_1 + 6.0NWBL_2$
w $17.5 + 8.0NWBL_1 + 4.5NWBL_2$
*instruction and destination in main, source in LCS, add 3.0, then add $6.0NWBL_2$
*instruction and source in main, destination in LCS, add 8.0, then add $10.5NWBL_1$.
instruction in main, data in LCS, add:
w/o $9.0 + 9.5NWBL_1 + 6.0NWBL_2$
w $7.5 + 8.0NWBL_1 + 4.5NWBL_2$

Class 2: Decimal Divide (DP).
instruction in LCS, data in main, add:
w/o 9.0
w 7.5
instruction and data in LCS, add:
w/o $20.5 + 5.5NWBL_2 + 9.0NWBL_1$
w $16.0 + 4.0NWBL_2 + 7.5NWBL_1$
instruction in main, data in LCS, add:
w/o $13.0 + 5.5NWBL_2 + 9.0NWBL_1$
w $11.5 + 4.0NWBL_2 + 7.5NWBL_1$
*instruction and destination in main, source in LCS, add $8.5 + 7.5NWBL_2$
*instruction and source in main, destination in LCS, add $10.5 + 9.0NWBL_1$

Class 3: Character Move (MVC, ZAP, PACK, UNPK) and Compare (CLC, CP).
instruction in LCS, data in main, add:
w/o 9.0
w 7.5
instruction and data in LCS, add:
w/o $18.5 + 6.0NWBL_2 + 6.0NWBL_1$
w $14.0 + 4.5NWBL_2 + 4.5NWBL_1$
instruction in main, data in LCS, add:
w/o $5.5 + 6.0NWBL_2 + 6.0NWBL_1$
w $4.0 + 4.5NWBL_2 + 4.5NWBL_1$
*instruction and destination in main, source in LCS, add $6.0 + 6.0NWBL_2$
*instruction and source in main, destination in LCS, add $6.0 + 6.0NWBL_1$

Class 4a: Translate (TR).
instruction in LCS, data in main, add:
w/o 9.0
w 7.5
instruction and data in LCS, add:
w/o $23.0 + 6.0NWBL_1 + 12.0N_2$
w $20.0 + 4.5NWBL_1 + 9.0N_2$
instruction in main, data in LCS, add:
w/o $9.5 + 6.0NWBL_1 + 12.0N_2$
w $9.5 + 4.5NWBL_1 + 9.0N_2$
instruction and destination in main, source in LCS, add:
w/o $3.5 + 4.0N_2$
w $4.0 + 3.5N_2$
instruction and source in main, destination in LCS, add:
w/o $2.0 + 6.0NWBL_1 + 4.0N_2$
w $2.0 + 4.5NWBL_1 + 4.0N_2$

Class 4b: Translate and Test (TRT).
instruction in LCS, data in main, add:
w/o 9.0
w 7.5
instruction and data in LCS, add:
w/o $23.0 + 6.0NWBL_1 + 5.5N_2$
w $20.0 + 4.5NWBL_1 + 4.5N_2$
instruction in main, data in LCS, add:
w/o $9.5 + 6.0NWBL_1 + 5.5N_2$
w $9.5 + 4.5NWBL_1 + 4.5N_2$
instruction and destination in main, source in LCS, add:
w/o $1.5 + 5.5N_2 - 2.0NWBL_1$
w $2.5 + 4.5N_2 - 2.0NWBL_1$
instruction and source in main, destination in LCS, add:
w/o $2.5 + 3.0NWBL_1$
w $2.5 + 1.5NWBL_1$

Class 5: All others:
instruction in LCS, data in main, add:
w/o 9.0
w 7.5

instruction and data in LCS, add:

w/o $24.5 + 6.0NWBL_2 + 12.0NWBL_1$

w $20.0 + 4.5NWBL_2 + 10.5NWBL_1$

instruction in main, data in LCS, add:

w/o $13.0 + 6.0NWBL_2 + 12.0NWBL_1$

w $11.5 + 4.5NWBL_2 + 10.5NWBL_1$

*instruction and destination in main, source in LCS, add $5.0 + 6.0NWBL_2$

*instruction and source in main, destination in LCS, add $10.0 + 12.0NWBL_1$

Interruption Times

A pending interruption condition (if not masked) is recognized and allowed to occur only at the termination of an instruction. The interruption time, then, extends from the time of this recognition to the beginning of the next instruction. Average interruption times (in microseconds) for the five classes of interruption are:

External	12	
Supervisor Call	12	
Program	12	
Machine Check	97	
I/O		
Device End	58*	41**
Channel End	28*	22**
Program Controlled (PCI)	28*	22**

*For multiplexer channel

**For selector channel

Legend for System/360 Timing

This section contains legends for the timing formulas for the cases where multiple timing formulas for instructions are listed. In some cases more than one timing formula for an instruction may be given.

Legends A_1 to A_4 are timing formulas for Store Multiple or Load Multiple instructions depending on quantity of general registers and position with respect to double word boundaries.

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 > 2 and 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

Legends B_1 to B_4 are timing formulas to be used when addressing a channel.

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

B_2: Use when addressing the multiplexer channel in the burst mode – first execution

B_3: (Same as B_2) – executions subsequent to the first, during the same burst mode operation

B_4: Use when addressing the selector channel

Legends C_1 to C_3 are timing formulas to use for radix (number base) conversion instructions, depending on the size of the number converted.

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 seven or less hexadecimal digits

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

Legends CH_1 to CH_4 are timing formulas to use for Compare Halfword instruction, depending on the nature of the numbers being handled.

CH_1: Use if signs differ

CH_2: Use if signs are alike, and the high-order 16 bits of the first operand are significant

CH_3: Use if inequality is found in byte 2

CH_4: Use if inequality is found in byte 3, or if comparison is equal

Legends CVD_0 to CVD_4 are timing formulas to use for the Convert to Decimal instruction, depending on the number of leading zero bytes.

CVD_0: Use if there are no leading zero bytes

CVD_1: Use if there is one leading zero byte

CVD_2: Use if there are two leading zero bytes

CVD_3: Use if there are three leading zero bytes

CVD_4: Use if there are four leading zero bytes

Legends D_1 to D_3 are timing formulas to be used depending on the state of the addressed channel.

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

D_2: Use if the multiplexer channel is busy and in the burst mode — first execution

D_3: (Same as D_2) — executions subsequent to the first, during the same burst mode operation

D_4: Use if the multiplexer channel is idle

D_5: Use if the multiplexer channel has an interruption pending

D_6: Use if the selector channel is busy

D_7: Use if the selector channel is idle

D_8: Use if the selector channel has an interruption pending

Legends E_1 to E_4 are timing formulas to use for the Execute instruction, depending on the instruction length code and varying conditions.

E_1: Use when subject instruction is one halfword long

E_2: Use when subject instruction is two halfwords long

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

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

Legends V_1 to V_4 are timing formulas to use for the Move instruction, depending on the location of operand fields.

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

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

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

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

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

This section contains the legends for terms to be used in the timing formulas for System/360.

ABV = Absolute value (i.e. unsigned value) of $NWBL_1 - NWBL_2$

B = Total number of bytes of the first operand which are processed (Applies to instructions with a single-length field)

E = Time for the subject instruction which is executed by the Execute instruction

ED = External delay

F = Input/output field length specified in Transfer I/O instruction

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

F_2 = 0 if the R_2 field (specified in the RR formatted branch instruction) is zero (i.e., branch is suppressed)
= 1 otherwise

G_1 = 1 if an overflow interruption occurs (PSW bit 36 = 1) or fixed-point divide interruption occurs
= 0 otherwise

G_2 = 1 if overflow occurs and fixed-point interruption is masked (PSW bit 36 = 0)
= 0 otherwise

G_3 = 0 if operand to be converted is positive
= 1 otherwise

G_4 = 1 if condition code is zero; i.e., all of the selected bits are zero or mask is all zero
= 0 otherwise

G_5 = 0 if first operand is positive
= 1 otherwise

GR = Number of general registers loaded or stored

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

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

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

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

$H_3 = H_2/2$ if H_2 is even
$= H_2/2 + 1$ if H_2 is odd

$H_4 = 4 - H/2$ if H is even
$= 4 - \frac{H-1}{2}$ if H is odd
(H_4 has a minimum value of 1)

K_1 = Number of zero-hexadecimal digits (both leading and embedded) in the factor (multiplier or multiplicand) with the smaller absolute value after recomplementing that factor if it is negative. In Multiply Halfword, K_1 applies only to the 16 low-order bits.

L_1 = 1 if a guard digit is involved
= 0 otherwise

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

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

M = greater of N_1 or N_2

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

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

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

N_1 = total number of bytes in the first operand (destination)

N_2 = total number of bytes in the second operand (source)

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 = number of field separator characters in edit pattern

N_5 = number of control characters in edit pattern

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

$NWBB_1$ = 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 (destination)

$NWBL_1L_2$ = 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 (source)

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

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

q_4 = quotient found by dividing the number of positions to be shifted by 4

q_8 = quotient found by dividing the number of positions to be shifted by 8

Q_4 = 1 if $q_4 = 0$
= 0 otherwise

QS = smaller of $N_1 - 8$ or $N_1 - N_2$

r_4 = remainder found after dividing the number of positions to be shifted by 4

R_3 = remainder when N is divided by 8

R_4 = 1 if $r_4 = 0$
= 0 otherwise

SG = number of signs in the field to be edited

S_1 = 1 if $r_4 = 3$, or if $q_4 = 0$
= 2 if $r_4 = 3$ and $q_4 = 0$
= 0 otherwise

S_2 = -1 if $r_4 = 0$
= 1 if $r_4 = 1$, and $q_4 = 0$
= 0 otherwise

S_3 = 0 if $r_4 = 0$, and $q_4 \neq 0$
= 1 if $r_4 = 0$, and $q_4 = 0$
= 3 if $r_4 = 1$
= 5 if $r_4 = 2$ or 3

S_4 = 0 if $r_4 = 0$
= 4 if $q_4 = 0$ and $r_4 = 1$, or if $q_4 \neq 0$ and $r_4 = 2$
= 3 if $q_4 = 0$ and $r_4 = 2$, or if $q_4 \neq 0$ and $r_4 = 3$
= 2 if $q_4 = 0$ and $r_4 = 3$
= 5 if $q_4 \neq 0$ and $r_4 = 1$

S_5 = 1 if the even-numbered register is zero
= 0 otherwise

S_6 = 1 if operand is negative
= 0 otherwise

S_7 = 1 if $r_4 \neq 0$ and operand is negative
= 0 otherwise

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

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

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

$T_{12} = 1$ if R_1 field of the Execute instruction is not zero
$= 0$ otherwise

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

$T_{14} = 1$ if $NWBL_2 = 0$
$= 0$ 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 $N_1 > \frac{N_2}{2}$
$= 0$ otherwise

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

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

$T_{20} = 1$ if signs are unlike for Add Decimal or if signs are alike for Subtract Decimal, when second operand > first operand
$= 0$ otherwise

U_1 = select out delay plus device delay

U_2 = device delay for Halt I/O sequence

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

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