IBM Field Engineering

Theory of Operation

(Manual of Instruction)

Restricted Distribution

This manual is intended for internal use only and may not be used by other than IBM personnel without IBM's written permission.

Specifications contained herein are subject to change without notice. Revisions and/or supplements to this publication will be issued periodically.

System/360 Model 44

Principles of Operation–Channels

IBM® Field Engineering

Theory of Operation

(Manual of Instruction)

Restricted Distribution

This manual is intended for internal use only and may not be used by other than IBM personnel without IBM's written permission.

Specifications contained herein are subject to change without notice. Revisions and/or supplements to this publication will be issued periodically.

System/360 Model 44

Principles of Operation—Channels

PREFACE

This volume describes the principles of operation of the input/output channels associated with the IBM 2044 Processing Unit, the processor for the IBM System/360 Model 44. The volume is divided into a Common Channel section which describes those areas of logic that are shared by both types of input/output channel; a Multiplexor Channel 0 section, which covers the operation of this channel and details the execution of input/output instructions; and a High Speed Multiplexor Channel section in which the operation of this channel and the execution of input/output instructions are covered.

The manual assumes knowledge of the System/360 as described in IBM System/360 Principles of Operation, Form A22-6821.

This volume is one of the manuals that constitute the IBM Field Enginering Theory of Operation manual for the IBM System/360 Model 44. The other volumes and their form numbers are:

System/360 Model 44, Introduction and Functional Units, Form Y33-0001: Gives a general outline to digital computers and computing technique, defines the relationship of the IBM 2044 to the System/360 and describes the various parts which form the processing unit.

System/360 Model 44, Principles of Operation - Processing Unit, Form Y33-0002: Describes the operation of the instructions for the accelerator and basic machine (other than floating-point instructions), and program and machine interrupts.

These volumes are referenced in other volumes by the main element of their titles.

Field Engineering Theory of Operation (FETO), IBM System/360 Model 44, Floating Point Feature, Form Y33-0005: Gives an introduction to floating-point arithmetic, describes the functional implementation of floating-point arithmetic in the 2044 and details the operation of floating-point instructions.

Field Engineering Theory of Operation (FETO), IBM System/360 Model 44, Single Disk Storage Drive, Form Y33-0006: Gives an introduction to the operation of the control unit and describes in detail the functional parts and the operations that may be performed.

Field Engineering Maintenance Manual (FEMM), IBM System/360 Model 44, Form Y33-0007: Contains information for servicing the 2044 Processing Unit.

Field Engineering Maintenance Diagrams (FEMD), IBM System/360 Model 44, Forms Y33-0008 and Y33-0009: Contains maintenance information in the following categories: Data Flow Charts, Flow Charts, Timing Charts, MAP's.

Other related manuals that describe units used in the System/360 Model 44 are:

Field Engineering Manual of Instruction (FEMI), 1052 Adapter, Form 223-2808.

Field Engineering Maintenance Manual (FEMM), Single Disk Storage/Direct Access, Form Y26-3663.

First Edition

This manual makes obsolete Field Engineering Theory of Operation, System/360 Model 44, Form Z33-0006-0.

The manual is written basically to Engineering Change Level 390049 and in some cases anticipates Engineering Change Level 390063. Significant changes or additions to the information in the manual will be covered in subsequent revisions or FE supplements.

This publication was prepared by IBM European Laboratories, Product Publications. A form is provided at the back of this manual for reader's comments. If the form has been removed, comments may be addressed to: IBM Corporation, FE Manuals, Dept. B96, PO Box 390, Poughkeepsie, N. Y. 12602

© International Business Machines Corporation 1966

CONTENTS (continued)

ILLUSTRATIONS

The following abbreviations are used in this manual.

CAR	Channel Address Register
CAW	Channel Address Word
CC	Chain Command (Flag)
CCW	Channel Command Word
CD	Chain Data (Flag)
CDR	Channel Data Register
COAR	Command Address Register
CPU	Central Processing Unit
CSW	Channel Status Word
DAR	Data Address Register
DEVAR	Device Address Register
HSMPX	High Speed Multiplexor (Channel)
IF	Interface
I-Fetch	Instruction-Fetch
IL	Incorrect Length (Latch, Flag)
I/O	Input/Output
IPL	Initial Program Loading
LUA	Load Unit Address
MPX	Multiplexor (Channel)
PCI	Program-Controlled Interruption (Flag)
PSW	Program Status Word
SAR	Storage Address Register
SDR	Storage Data Register
SLI	Suppress-Length-Indication (Flag)
STAN	Status Analysis (Latch)
TIC	Transfer-in-Channel
UCW	Unit Control Word
μs	Microsecond

COMMON CHANNEL

The common channel is an area of hardware containing controls that are shared by all channels. These controls, as shown in Figure 3B-1, are:

1. An Input/Output (I/O) clock, that is used to generate timing pulses for the channels.
2. A channel-decode circuit, that is used to select the correct channel during an I/O or Initial Program Loading (IPL) operation.
3. Channel condition code latches, that are used to set the condition code in the Program Status Word (PSW) from any channel.
4. An I/O cycle request circuit, is used to set the I/O cycle control from any channel, and a priority circuit that is used to decide which channel requested an I/O cycle.

I/O CLOCK

- Uses the main clock in CPU as a sync pulse.
- Is continuously running.
- Generates the I/O clock pulse A, B, C and D.
- I/O clock pulses are used as timing pulses for the channels.

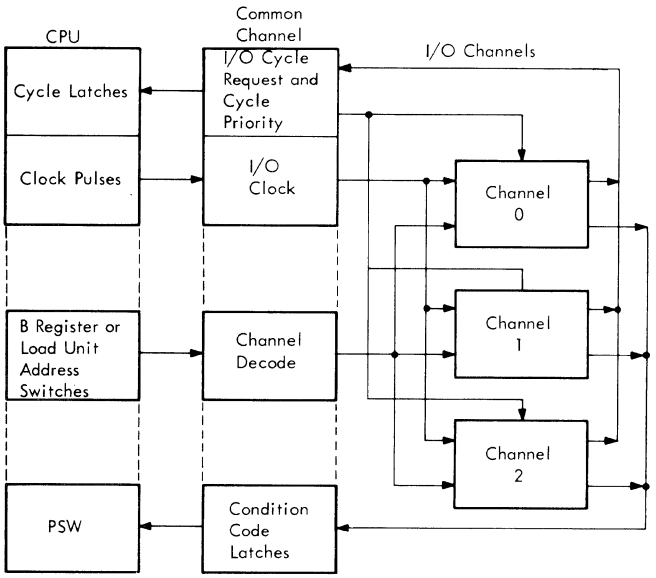
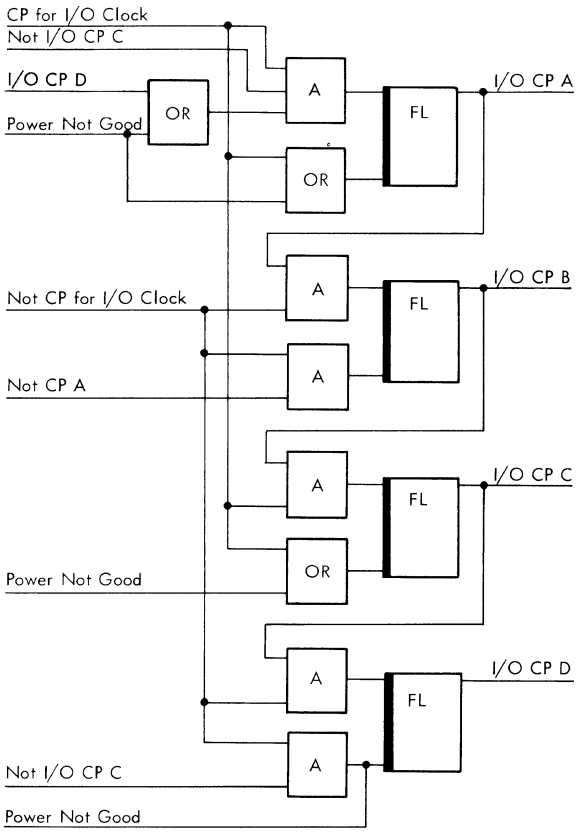


Figure 3B-1. Common Channel

Figure 3B-2 shows the logic of the I/O clock. A clock pulse from the basic clock in the Central Processing Unit (CPU) is used to drive the I/O clock. The I/O clock consists of four latches: I/O clock A, B, C and D latches.



Refer also to ALD pages FA 101 to 111

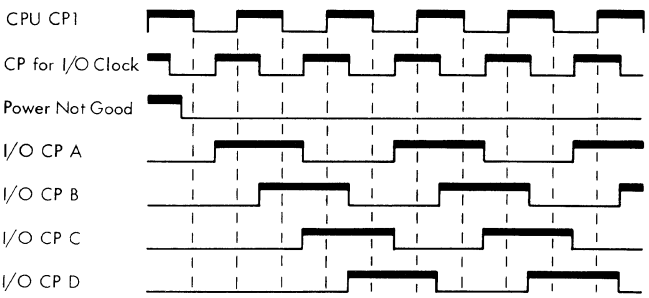


Figure 3B-2. I/O Clock Latches

The I/O clock A latch is turned on as soon as power is good and when the first 'clock pulse for I/O clock' appears. The I/O clock B latch is set when 'clock pulse for I/O clock' ends. The I/O clock B latch turns on the I/O clock C latch which, in turn, sets the I/O clock D latch.

The output of the I/O clock D latch is then gated back to the I/O clock A latch and this sequence of operation continues until either power is not good or the 'clock pulse for I/O clock' stops.

The timing and duration of the I/O clock pulses is shown in Figure 3B-2.

CHANNEL-DECODE CIRCUIT

- Used to select one subchannel during any I/O operation.
- Used to check if the channel address is invalid.

The logic associated with the channel-decode circuit is shown in Figure 3B-3. This circuit is used to select either Multiplexor (MPX) channel 0, High Speed Multiplexor (HSMPX) channel 1 or HSMPX channel 2 during any I/O operation. Bits 22 and 23 in the B register are used to decode the channel for a start I/O, test I/O, halt I/O or test-channel operation. During an IPL operation, however, the channel is decoded from the bits set up by the Load Unit Address (LUA) switches on the console.

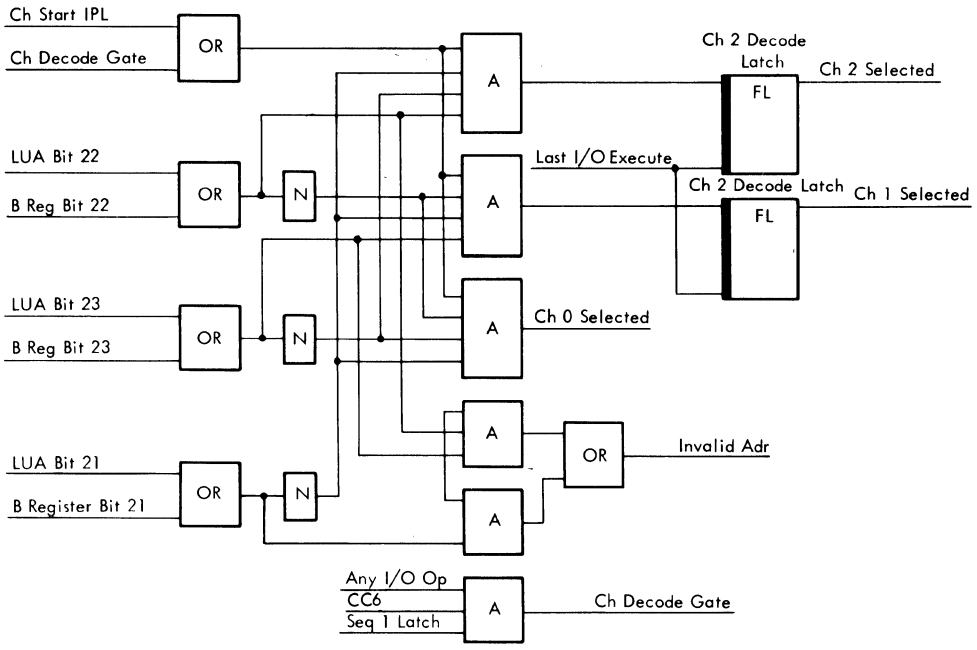
The channel selected according to bits 22 and 23 is shown in the table in Figure 3B-3.

A one bit in both positions 22 and 23 represents an invalid address. Also, bit 21 must always be off in order to prevent an invalid address. If an invalid address is detected, no channels are selected, and the current I/O operation is terminated.

CHANNEL CONDITION CODE LATCHES

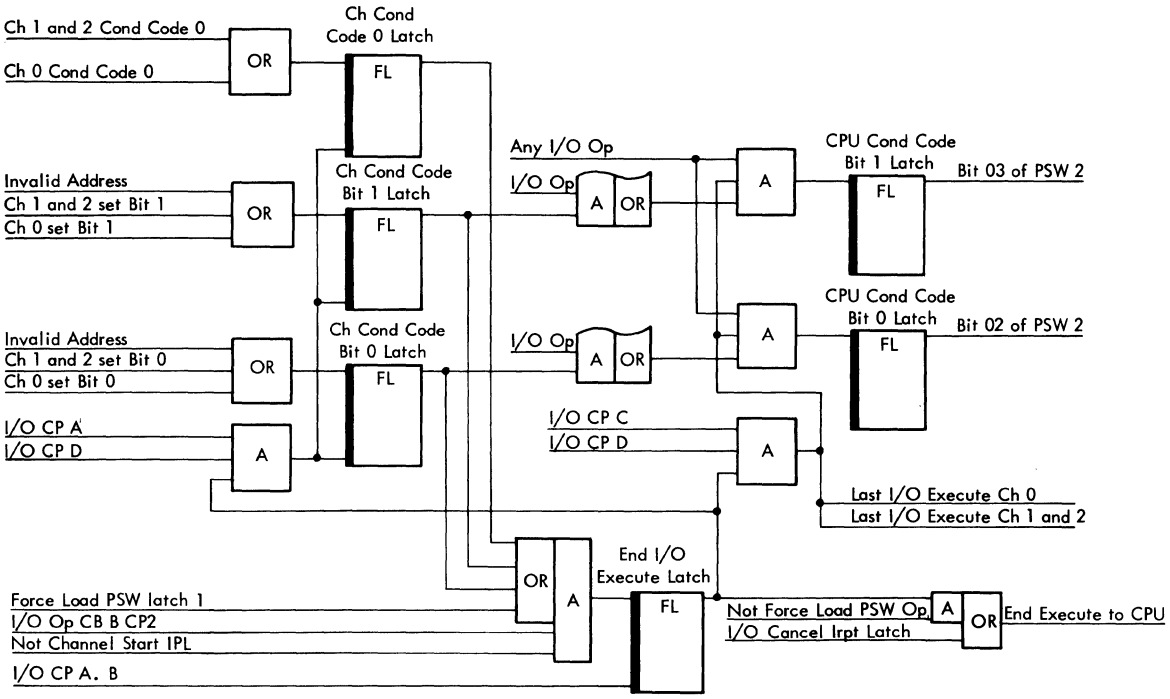
- Used to set the condition code in the PSW.
- Used to end the execution of any I/O operation.

The channel condition code latches store any condition code set from any channel into the PSW and, at the same time, terminate the current I/O operation. There are three channel condition code latches as shown in Figure 3B-4. The channel condition code bit 0 and bit 1 latches identify the condition code bits to be stored in the PSW and are used to set the 'end I/O execute' latch. The channel condition code 0



B Register or LUA Bits			Channel Selected
21	22	23	
0	0	0	Channel 0
0	0	1	Channel 1
0	1	0	Channel 2
0	1	1	Invalid

Figure 3B-3. Channel Decoding



Channel and CPU Cond Code Latches			
Bit 0	Bit 1		
0	0	0	00
0	1	1	01
1	0	2	10
1	1	3	11

Figure 3B-4. Channel Condition Code Latches

latch is used only to turn on the 'end I/O execute' latch.

When any of these three condition code latches is set from a channel, the 'end I/O execute' latch is turned on at I/O clock pulse B, clock pulse 2. An 'end execute' signal is then sent to the CPU in order to fetch the next instruction. At I/O clock pulses C and D, the contents of the channel condition code bit latches are gated to the CPU condition code bit latches (bits 02 and 03 of PSW 2). The same signal is also sent to the channels in order to turn off the execute latches. During the next I/O clock pulses A and D, the condition code latches are reset. The 'end I/O execute' latch is reset at the following I/O clock pulses D and A.

During an IPL operation, the 'end I/O execute' latch is inhibited from being set until the 'force load PSW' latch is set. Refer to the IPL description for the respective channel in this manual.

CYCLE PRIORITY AND I/O CYCLE REQUEST

The channels use I/O cycles in order to fetch or store data in main storage. An I/O cycle may be requested from any channel at any time. However, only one channel may use the I/O cycle at one time. A priority circuit among the channels is therefore necessary. All channels use the same circuit to set the I/O cycle control latch.

Channel Priority

- The I/O cycle has priority over all CPU cycles.
- The HSMPX channels have priority over MPX channel 0.
- The priority between the two HSMPX channels is equally shared.
- The priority between the subchannels is on an A, B, C, D basis.

An I/O cycle always has priority over a CPU cycle, except for double cycles. This priority is achieved by setting the I/O cycle control latch one clock pulse earlier than the CPU cycle control latches.

The HSMPX channels have priority over MPX channel 0. MPX channel 0 can therefore take an

I/O cycle only if no I/O cycle request is up from the HSMPX channels.

The priority between the HSMPX channels is shared equally. If the two channels have a request up at the same time, the one that was not serviced on the last occasion will have the higher priority.

An I/O cycle request latch is provided for each subchannel, and more than one latch may be set at the same time. The priority between the subchannels is on an A, B, C, D basis: for example, subchannel A has top priority and subchannel D has the lowest priority. An I/O cycle request for data service interrupts a chaining routine if the data service is requested from a subchannel with a higher priority. A data service request for a chaining operation generated from any subchannel also interrupts a Channel Command Word (CCW) fetch for start I/O. However, only one subchannel may perform a chaining routine at the one time. Figure 3B-5 shows an example of the I/O cycles taken for the different channels and subchannels.

MPX Channel 0 I/O Cycle Request

- I/O cycles for data service.
- I/O cycles to store and fetch Unit Control Words (UCW's).
- I/O cycles to fetch the CCW.

The logic associated with the MPX channel 0 I/O cycle request is shown in Figure 3B-6. The conditions that may set the I/O cycle request are listed above.

HSMPX Channel I/O Cycle Request

- I/O cycles to fetch the CCW.
- I/O cycles for data service.

The HSMPX channel uses I/O cycles to fetch the CCW for a start I/O operation and for a chaining operation. I/O cycles are also used to store or fetch data.

The logic associated with the HSMPX channel I/O cycle request is shown in Figure 3B-7. Each subchannel has two groups of I/O cycle request latches. One group is set if data service is required and the other group is set for a chaining operation. If any of these latches is on, an attempt is made to set the subchannel priority request latch. This latch may be set at any time if the I/O request A and C latches are off, but only at WC1, CP1, if the I/O request A and C latches are already set. The outputs from the subchannel priority request latches are OR'ed together to give a channel 1 or channel 2 request.

The CCW latch is used to indicate that a CCW is to be fetched for a start I/O operation for either channel; the latch is also OR'ed with the outputs of the channel 1 and channel 2 request and the output is the final HSMPX I/O cycle request.

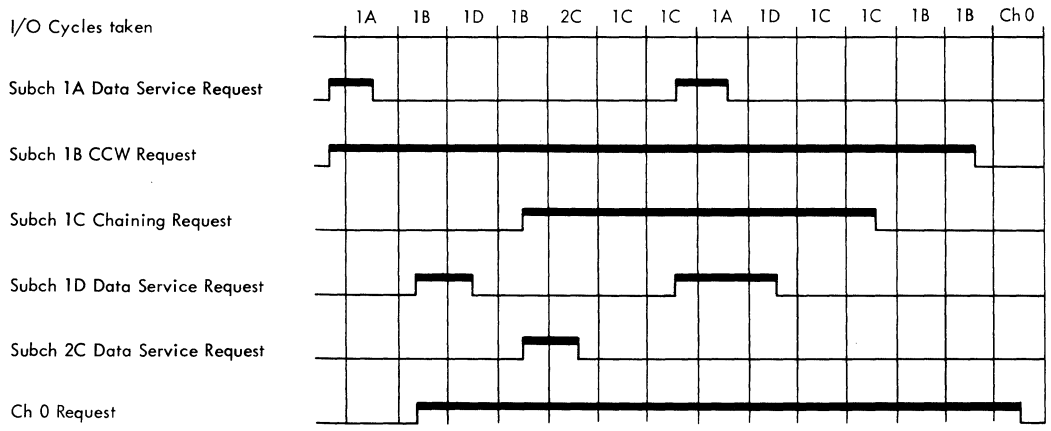


Figure 3B-5. Example of I/O Cycles taken for the Different Channels

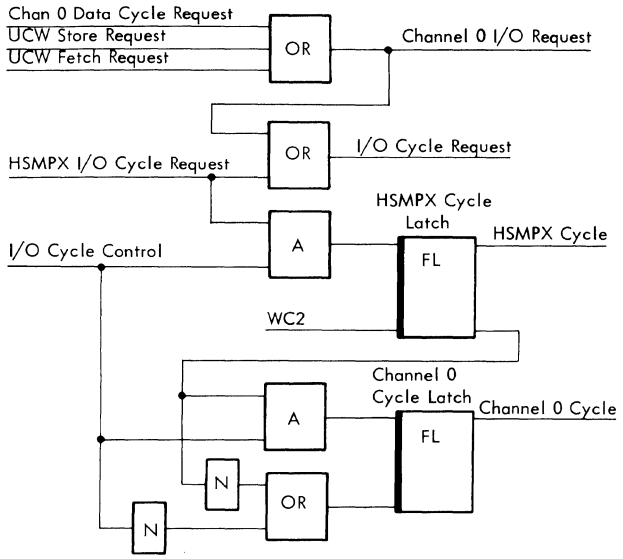


Figure 3B-6. Channel 0 I/O Cycle Request and Channel Cycle Latches

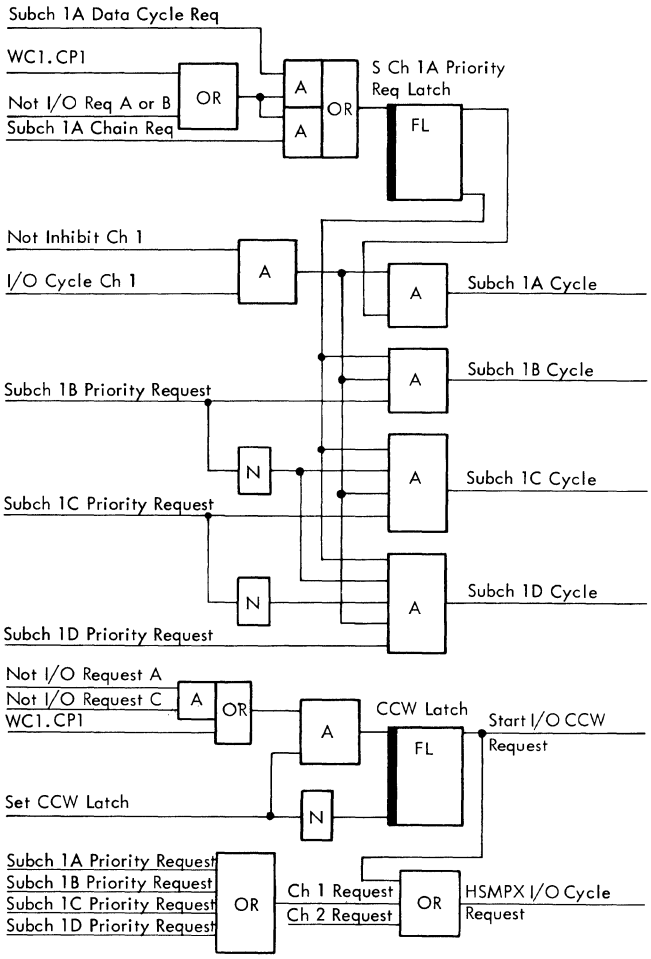


Figure 3B-7. HSMPX Channel I/O Cycle Request

I/O Cycle Request Latches

- Used by all channels to set the I/O cycle control latch.
- Generates a delay from the time the request appears to the start of the read/write clock.
- The priority between the channels and subchannels is established during this delay.

The circuit used to set the I/O cycle control latches is shown in Figure 3B-8. The circuit is used by all channels and consists of two request latches and two I/O cycle control latches. Between the time that an HSMPX channel raises a request and the read/write clock is started, the priority must be established; the read/write clock must therefore not be started immediately after the cycles are requested. The I/O cycle control latches may be set either during a read/write cycle or at any other time.

I/O Cycle Request With No Read/Write Cycle

- The channel that first raised the request is granted the I/O cycle.

If any channel raises a request for an I/O cycle and the read/write clock is off, the channel that raises the request is granted the I/O cycle, even if a channel with a higher priority sends a request soon afterwards.

Assume that the request is raised at I/O clock pulse C (CP C). The I/O request C latch is set, thus preventing any other subchannels from setting their priority request latches. At I/O CP A, CP1, the I/O cycle control A latch is set, provided that no CPU cycle control latch is on. The read/write clock is started at the next I/O pulse. The sequence of this operation is shown in Figure 3B-9. The delay indicated is necessary in order to establish the priority.

If the request is raised at I/O CP A time, the I/O request A latch and I/O cycle control C latch are set instead of the I/O request C and I/O cycle control A latches.

I/O Cycle Request During a Read/Write Cycle

- The subchannel priority request latches are sampled at Write Clock 1 (WC1).
- Any request detected at WC1, CP1 or before, gives an I/O cycle as the next cycle.

If no I/O cycle request is up at the beginning of a read/write cycle, the channel or subchannel that

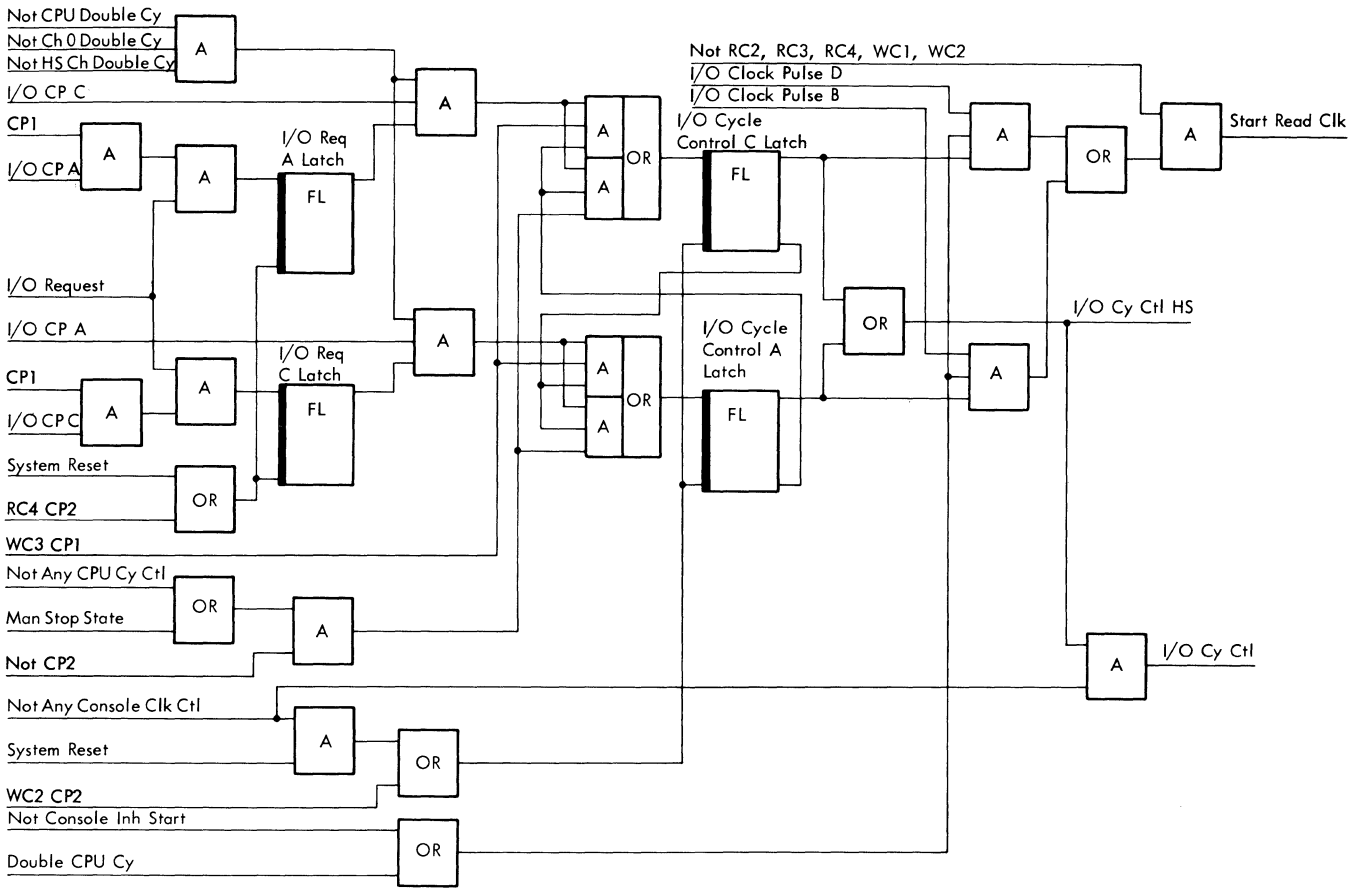


Figure 3B-8. I/O Cycle Request Latches

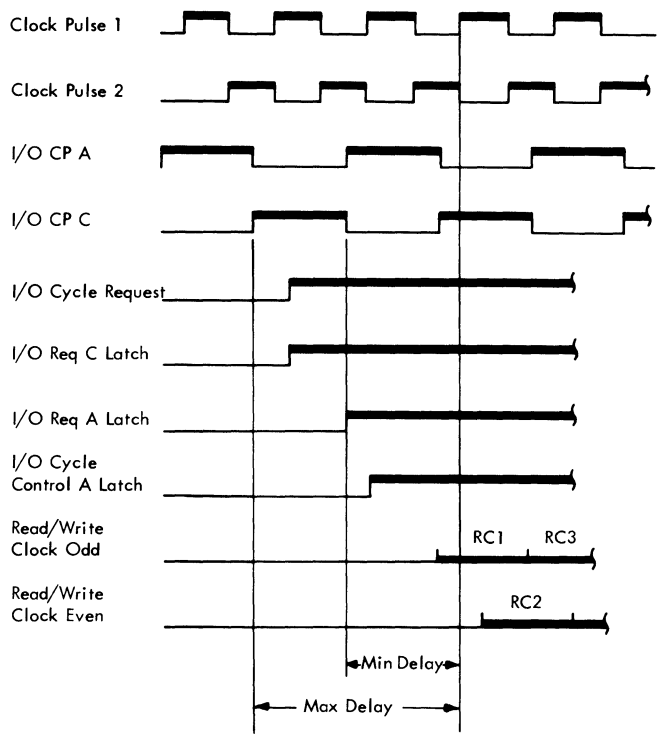
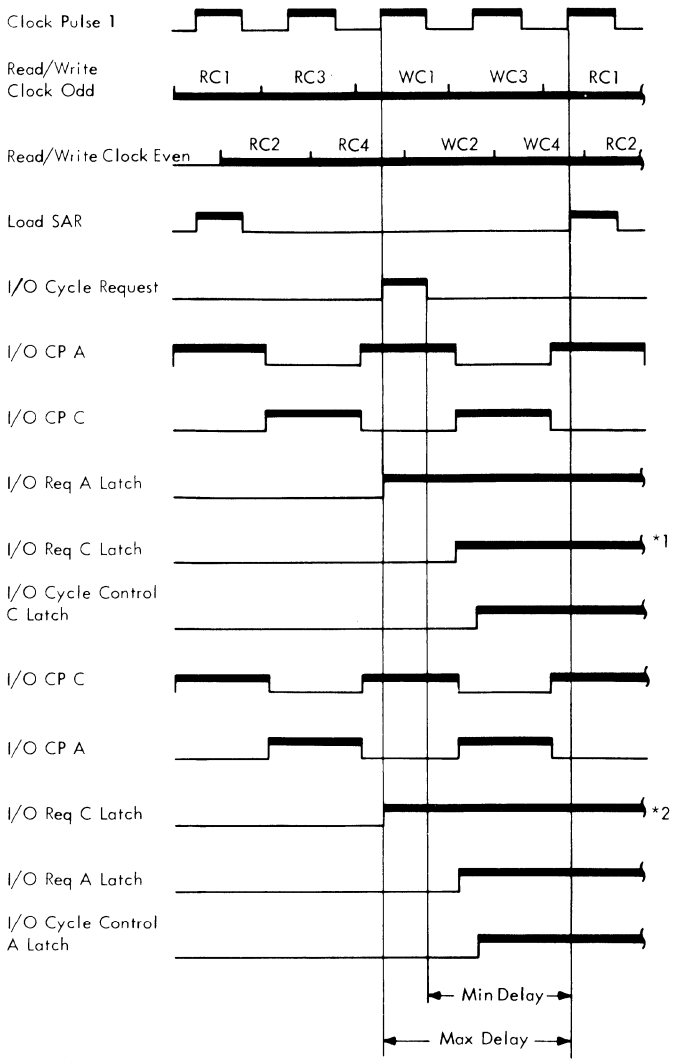


Figure 3B-9. I/O Cycle Request with no Read/Write Cycle

first detects a request for an I/O cycle sets the I/O request A and C latches. This prevents any other subchannel priority latches from being set. At WC1, CP1, the subchannel priority request latches may again be set if their respective request latches are on (see Figure 3B-7.) After WC1 however, the state of the subchannel priority request latches cannot be altered and the priority is then established.

Assume that I/O CP A occurs at CP1 and that a request is detected at the same time. The I/O request A latch is set. At the next I/O clock pulse, the I/O cycle control C latch is set at WC2, CP1, thus ensuring that the I/O cycle control latch is always set at the beginning of clock pulse 1 (CP1). The read/write clock is started at WC4. The sequence of this operation is shown in Figure 3B-10. If I/O clock pulse C occurs at the same time as WC1, the I/O request C and I/O cycle control A latches are set instead of the I/O request A and I/O cycle control C latches.



*1 If I/O CP C occur together with WC 1
*2 If I/O CP A occur together with WC 1

Figure 3B-10. I/O Cycle Request during a Read/Write Cycle

Channel and Subchannel I/O Cycles

Although an I/O cycle may be requested first by one channel, the resulting I/O cycle may still be taken by another channel; this depends on the state of the request latches for the three channels and the priority among them.

MPX Channel 0 I/O Cycles

- MPX channel 0 I/O cycles are taken if no requests are present from the HSMPX channels.

HSMPX Channel I/O Cycles

- An HSMPX channel 1 cycle is taken unless channel 1 is inhibited by channel 2.
- An HSMPX channel 2 cycle is taken unless channel 2 is inhibited by channel 1.
- The subchannel cycle taken depends on the subchannel priority request latches and their priority.

If an I/O cycle is requested from both channels at the same time, one of the channels must be prevented from taking the cycle. The circuit for channel priority is shown in Figure 3B-11. Each time HSMPX channel 1 has an I/O cycle, the channel 1 cycle latch is set. This latch turns on the channel 1 inhibit latch at WC2. Therefore, HSMPX channel 1 is inhibited and HSMPX channel 2 takes the first I/O cycle.

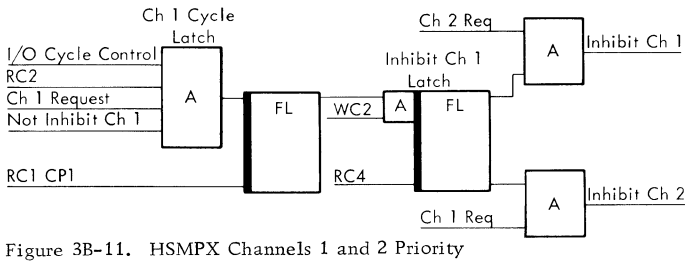


Figure 3B-11. HSMPX Channels 1 and 2 Priority

When either of the HSMPX channels has been granted an I/O cycle, the subchannel priority request latches are tested (Figure 3B-7). The subchannel with the highest priority takes the I/O cycle; for example, if subchannel A priority request latch is on, all other subchannels are prevented from taking the cycle.

When the subchannel priority is established, the address in the selected subchannel may be gated to the Storage Address Register (SAR), as shown in Figure 3B-12.

An I/O cycle for a start I/O CCW-fetch is taken if no subchannel priority request latches are on.

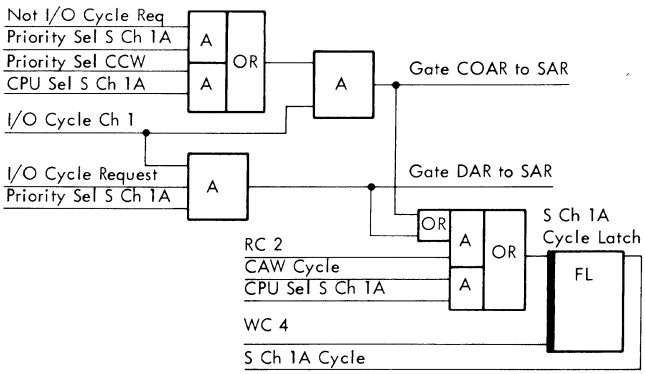


Figure 3B-12. Subchannel Cycle Latch

MULTIPLEXOR CHANNEL 0

- An I/O channel controls the transfer of information between the CPU and the I/O devices.

The channels are used to transfer information between the CPU and its I/O devices. When the CPU decodes an I/O operation, it loads into the channel the CCW. As soon as the CCW is loaded, the CPU is released and the channel takes over control of the I/O operation. The CPU then proceeds with the next sequential instruction. When the channel either receives or requires a byte of data it requests access to main storage. This is the only interference with the CPU operation until the channel has completed the current operation, at which time it will refer back to the CPU.

DESCRIPTION

- Channel 0 is the standard-speed multiplexor channel of the IBM 2044.
- The channel has 32 or 64 subchannels.
- Channel consists of hardware registers.
- Subchannels are areas of extension storage.

MPX channel 0 is the standard-speed multiplexor channel of the IBM 2044 processing unit. Depending upon the size of the main storage of the system, the channel may have either 32 or 64 subchannels. The channel consists of a number of hardware registers and their associated control circuitry. The subchannels are areas of extension storage.

MPX channel 0 is capable of handling only one byte of data at a time. It therefore requests access to main storage for each byte that is received from or sent to an I/O device. A block diagram of the channel data flow is given in Figure 3B-13.

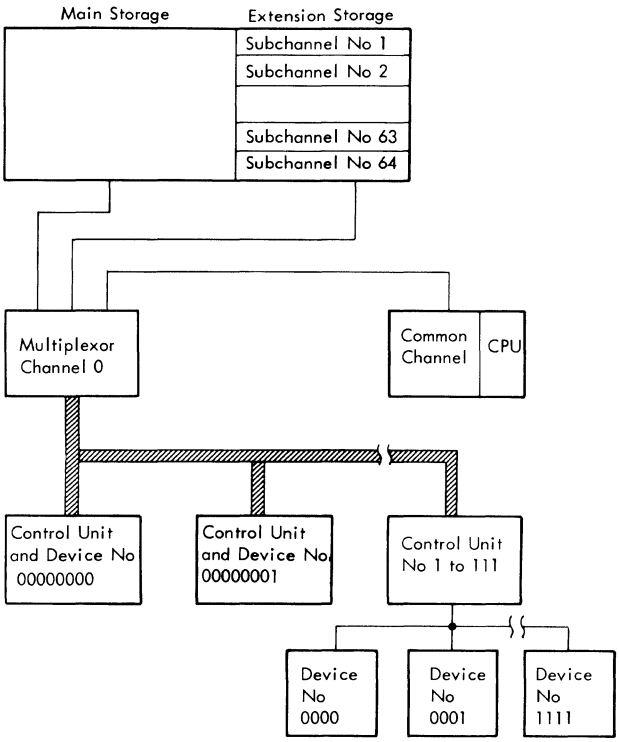


Figure 3B-13. MPX Channel 0 Data Flow Block Diagram

Performance

- Channel 0 operates either in byte or burst mode.

When a channel 0 operation is decoded by the common channel, the CCW for that operation is fetched from main storage. If the required device is available, the contents of the CCW are stored in an area of extension storage, the address of which will depend on the address of the device. These areas of extension storage are known as UCW's. Each time a service request is received from the device, the UCW's for that subchannel are fetched from extension storage and loaded into the channel registers where they remain until the data byte has been transferred from or to the device. The updated information is then loaded back into extension storage. This process allows the channel to control several I/O operations at the same time. If, however, the device currently operating on the channel requires a faster data transfer rate, the UCW's remain in the channel registers from the first data service until the operation is complete. There are, therefore, two modes of operation, known respectively as byte mode or burst mode.

Priority

- Channel 0 has the lowest priority of the channels.

Because MPX channel 0 is of a lower speed than the HSMPX channels, it has the lowest priority of the three channels. Therefore, if either of the HSMPX channels is operating, channel 0 is able to access main storage only during the time when the HSMPX channels are not requesting main storage access. The channels, however, have a higher priority than the CPU. The channel accesses main storage during either I/O cycles or C-cycles.

Capacity

- Channel 0 has either 32 or 64 subchannels.
- A maximum of eight control units may be connected to the channel.

Channel 0 can have either 32 or 64 subchannels, depending on the storage capacity of the CPU, as follows.

Model E (32K): 32 subchannels.
Model F (64K): 64 subchannels.
Model G (128K): 64 subchannels.

Because of the electrical characteristics of the standard interface, a maximum of eight control units can be connected to channel 0. Each control unit may, in turn, be connected to a maximum of 16 devices.

CHANNEL DATA FLOW

MPX channel 0 data flow is shown in FEMD Figure 1002 while the data transmission control circuitry is shown schematically in Figure 3B-14 of this manual.

The data flow to and from the CPU Storage Data Register (SDR) is 32 bits wide (full word) or four bytes, with no parity. Since MPX channel 0 handles one byte of information at a time, a word is called four times and the required byte is selected. When a byte of data is read from a device, the position of the byte in the word must also be controlled (Figure 3B-14). However, a complete 32-bit word is transferred from the SDR to the five control registers in the channel when they are being loaded, and also when they are being unloaded back to SDR for transfer to their subchannel.

The five registers which control the operation to be performed are the 'command', D, E, 'flags' and 'channel status' registers.

Command Register

- Six bits wide.
- Bits are numbered 2 (leftmost) to 7.
- Holds two "housekeeping" flags and four bits of the command byte.

This six-bit register contains only the low-order bits of the command byte in positions 4 to 7. The entire command is placed in the Channel Data Register (CDR) for transmission to the device.

The other two bits of the command register, bits 2 and 3, are the 'halt' flag and 'end' flag respectively. The 'halt' flag indicates that a halt instruction could not be obeyed, and that a stop will be issued to the device whenever it calls over the interface.

The 'end' flag is set whenever the device submits a 'channel end' to the channel. If the channel-end condition is accepted in the channel, bits 4 to 7 (command) are reset; if the 'channel end' is stacked in the device, these command bits are retained. The 'end' flag stays on until the channel-end interrupt is accepted by the CPU. The command bits therefore indicate whether the interrupt is still in the device or has been accepted.

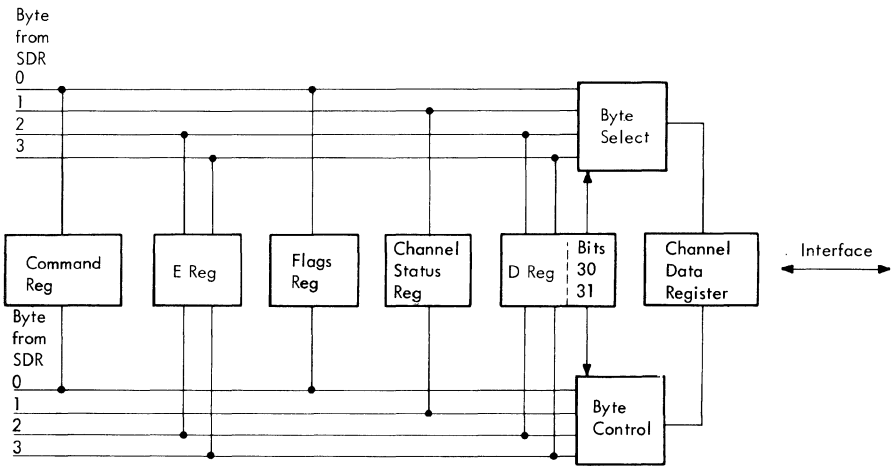


Figure 3B-14. Data Transmission

D Register

- Eighteen bits wide.
- Bits are numbered 14 (leftmost) to 31.
- Normally holds data address.
- On initial loading is used to develop the next command address.

This 18-bit register normally contains the address of the next data-word location in main storage. The two low-order bits (30 and 31) are not used to address storage. Addresses in the SAR are therefore confined to word boundaries. Bits 30 and 31 are used to control byte selection within a word at the channel; this determines which byte is read from the device to main storage (byte control) or which byte is transmitted to the device from main storage (byte select). The D register is also used during initial selection to develop CCW addresses (both the current and the next sequential, CCW + 8). These two addresses (CCW and CCW + 8) are stored in UCW 3 and, should command chaining or data chaining be specified, may be extracted by the channel controls without recourse to the original instruction which may have been destroyed.

The address field uses bits 08 to 31 of PSW 2 in the CPU. Of these, bits 08 to 14 are unnecessary for machines with a main storage capacity of up to 131,072 (128K) bytes. These high-order bits are therefore compressed into one bit (14) in the register. This bit is sufficient to indicate if the address specified is outside storage limits. The compressed bit is sent to the SAR with bits 15 to 29 and the CPU Invalid Memory Address (IMA) logic signals the channel if bit 14 is set, thus preventing any processing of data.

E Register

- Eighteen bits wide.
- Bits are numbered 14 (leftmost) to 31.
- Normally holds byte count.
- On initial loading is used to store the command address.

This is an 18-bit register. Bits 8 to 14 are compressed in a similar manner as for the D register. The E register normally holds the byte count which is reduced by one at the end of each data cycle. A zero-detect circuit continuously monitors bits 16 to 31 and, if a data cycle is requested when the count is zero, a stop signal is issued to the device.

During initial selection, the E register also holds the Channel Address Word (CAW) which is the CCW address. The E register uses this CCW 1 address to load the SAR, and the contents of addressed location are then placed in the command and D registers. The E register is then incremented by four to give the address of CCW 2 which is sent to SAR. The E register is cleared and the contents of CCW 2 are read out of main storage into the E register, flags register and channel status register.

The E register is equipped with a decrement circuit but no increment circuit. Therefore, to create the condition of incrementing CCW 1 by four, a bit is injected into bit position 29.

CCW's must always be situated in main storage on double-word boundaries to ensure that bits 29 to 31 are zero.

Flags Register

- Five bits wide.
- Bits are numbered 0 (leftmost) to 4.
- Holds flag bits from CCW 2.

This five-bit register holds the five flags transferred from CCW bits 32 to 36. The register positions are as tabulated.

Flag Bit	CCW Bit	Description
0	32	Chain-Data (CD) flag.
1	33	Chain-Command (CC) flag.
2	34	Suppress-Length-Indication (SLI) flag.
3	35	Skip (SKIP) flag.
4	36	Program-Controlled-Interruption (PCI) flag.

Channel Status Register

- Six bits wide.
- Bits are numbered 8, 9, 10, 12, 13 and 14.
- Holds channel status bits.

This six-bit register stores the indications of special conditions occurring within the channel and supplies details to the program via the Channel Status Word (CSW). The register conditions are:

Bit 8 : Channel status PCI.
Bit 9 : Channel status incorrect length.
Bit 10: Channel status program check.
Bit 12: Channel status data check.
Bit 13: Channel status control check.
Bit 14: Channel status interface control check.

Channel Address Register

- Eight bits wide.
- Holds device address.

The channel address register (CAR) holds the device address of eight bits. On initial selection, the address is taken from the B register or LUA switches When the device is selected, the address is gated via the standard interface to the device.

At all other times, the CAR is set from the device with an 'address in' via the standard interface and is decoded for use as the UCW (subchannel) address in extension storage.

Channel Data Register

- Eight bits wide.
- Holds the command byte on initial selection.

This eight-bit register is used on initial selection to hold the command byte. When the device is selected, the CDR is gated to the standard interface with 'command out'. When 'status in' rises, the register is reset, then set with the status byte. The status may be analyzed independently with respect to the data flow.

During data cycles, the register is set with 'service in'. If the device is operating in write mode, the register is cleared and then set with the data from the SDR.

Parity is not checked in the multiplexor channel 0, but a byte on the standard interface travelling towards the device has a parity bit generated. The parity of a byte travelling from the device is checked on the interface prior to arrival at the channel. Incorrect parity is signalled by data or interface control check.

Interrupt Buffer Register

- Eight bits wide.
- Slave of the CAR.
- Set from the CAR when an interrupt is pending in the channel.

The eight-bit register is a slave of the CAR. Whenever an interrupt is pending in the channel, the interrupt buffer register is set from the CAR. A set of latches signals to the CPU the type of interrupt. When the CPU is ready to accept the interrupt, the interrupt buffer register is used to address the subchannel in extension storage.

Provision of this register allows other devices to be operated between the occurrence of the interrupt and its acceptance by the CPU. However, only one interrupt can be stored in this way within the channel.

The interrupt buffer register also controls the clearing of extension storage after a channel check has been accepted by the CPU.

The register normally addresses extension storage. With the inclusion of a decrementing control and by setting the highest device address, successive storage cycles read out the extension storage words and, by blocking the entry to the SDR, all zeros are read back with good parity.

When the interrupt buffer register reaches zero address, a 'channel 0 clear' signal is developed. This ends the operation.

Address Match

- Is not a register but comprises a group of comparison circuits.

The address match (shown in FEMD Figure 1002) is not a register but a group of circuits that makes three address comparisons, all with the CAR. The other addresses are gated through a common funnel. These comparisons are made:

1. When the device response address on 'bus in' is compared with the CAR to determine whether the correct device responded to initial selection.
2. When the interrupt buffer register is compared with the CAR to determine whether the device called has an interrupt pending in the channel.
3. When the address of a device that submits a stacked 'channel end' is entered into UCW 1 for future reference. If that subchannel is to be tested by the test I/O operation, UCW 1 is fetched and the address is compared. If a match is obtained, the interrupt is cleared; otherwise the 'subchannel busy' signal is given.

Generate Extension Storage Address

- Is not a register but is an area of logic circuitry.
- Converts eight-bit device address to 16-bit subchannel address.

The generate extension storage address (shown in FEMD Figure 1002) is not a register but an area of logic circuits. The circuitry converts a device address of eight bits into a subchannel address of 16 bits.

Two device-address formats are available as shown in Figure 3B-15. In a single subchannel format, the most-significant bit is a zero to indicate a single subchannel. The next bit must also be a zero. The remaining six bits specify the subchannel for that device.

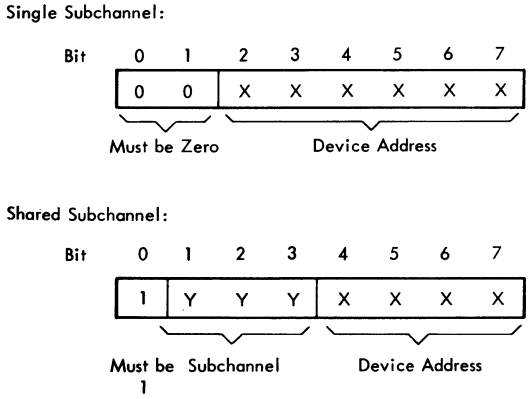


Figure 3B-15. Device-Address Formats

In the format for shared subchannel devices, the most-significant bit is a one to indicate a shared subchannel. The next three bits indicate the subchannel. The limit is therefore eight subchannels. The last four bits specify the device on that subchannel. Thus, 16 devices can be attached to a control unit.

When addresses are translated, the subchannel bits of a shared subchannel address are switched to the low part of a single subchannel address. Thus addresses 0000101 and 11011010 call for the same UCW. After the bits have been translated, the entire SAR address is inverted so that 00110101 becomes:

0 1 1 1 1 1 1 1 1 1 1 0 1 0 X X Y Y

14 31

Note: Bit 14 is zero, since a one would be invalid.
Bits 15 to 19 are forced ones.
Bits 20 to 27 are the inverted device address.
Bits 28 and 29 are used to address the required UCW.
Bits 30 and 31 are not used, as they address bytes.

Unit Control Words

- UCW's are areas of extension storage.

The UCW's are used to store the control information for an I/O operation between data transfers. Three words are required to hold the necessary data but, for convenience in addressing, UCW's are stored in groups of four. The format of the UCW's is shown in Figure 3B-16.

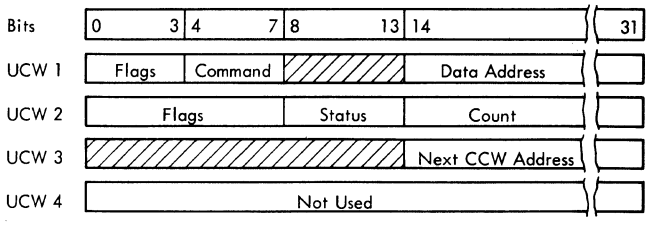


Figure 3B-16. UCW Format

TYPES OF CYCLES

The types of cycles associated with MPX channel 0 operations are shown in Figure 3B-17.

Type of Cycle	Storage Request	Data Source	Address Source
I/O	UCW Fetch Word 1	Extn Stor and SDR	CAR
I/O	UCW Fetch Word 2	Extn Stor and SDR	CAR
I/O	UCW Fetch Word 3	Extn Stor and SDR	CAR
I/O	UCW Store Word 1	Command and D Registers	CAR
I/O	UCW Store Word 2	Flags and E Registers	CAR
I/O	UCW Store Word 3	D Register	CAR
C	Ctl Wd Fetch CAW	Main Storage and SDR	Common Chan Addr Gen
I/O	Ctl Wd Fetch Word 1	Main Storage and SDR	E Register
I/O	Ctl Wd Fetch Word 2	Main Storage and SDR	E Register
C	Ctl Wd Store	Unit Status (CDR), Channel Status Register	Common Chan Addr Gen
C	UCW 1 Reset Cycle	SDR (zeros)	Interrupt Buffer Register
C	CSW 1 (double) Cycle	UCW to SDR	Common Chan Addr Gen Interrupt Buffer Register
C	CSW 2 (double) Cycle	UCW to SDR	Common Chan Addr Gen Interrupt Buffer Register
I/O	Data Cycle (read)	Channel Data Register	D Register
I/O	Data Cycle (write)	SDR	D Register

Figure 3B-17. Types of Cycles

START I/O OPERATION

- Start I/O is used to begin all I/O data-transfer operations.

The flow charts and timing charts for the start I/O operation are contained in FEMD Figures 6101 to 6104. The start I/O instruction is used by the CPU to commence a channel data-transfer operation. The operation first checks that the channel is free. The information contained in the CCW is then transferred to the channel and the control unit and device are selected. If they are free, the command is sent to the control unit. At this point, the CPU is released and the contents of the channel registers are loaded into the UCW's associated with that subchannel.

Instruction-Fetch

- The start I/O instruction is decoded by the CPU.
- The channel, subchannel, control unit and device address is generated in the B register.

The start I/O instruction uses the SI format. The CPU must be in the supervisor state when the instruction is executed.

The Instruction-Fetch (I-fetch) phase is described in Principles of Operation - Processing Unit, Form Y33-0002, while flow charts and timing charts are given in FEMD Figures 6001 to 6006 and 6201 to 6206.

At the end of I-fetch, the address of the channel, subchannel, control unit and device is contained in the B register bits 21 to 31. The address format varies depending on whether one or two control units are associated with the subchannel. The formats for single subchannel and for shared subchannel are shown in Figures 3B-18 and 3B-19 respectively; note that bit 21 (in both formats) and bits 24 and 25 (single subchannel format) are not used and each must be zero.

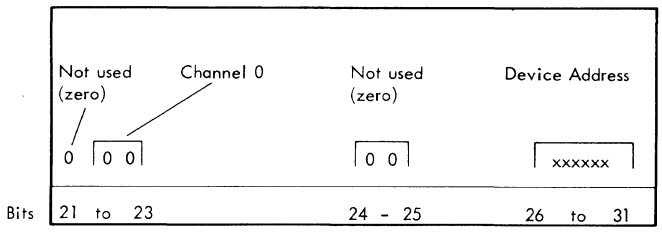


Figure 3B-18. Single Subchannel Address Format

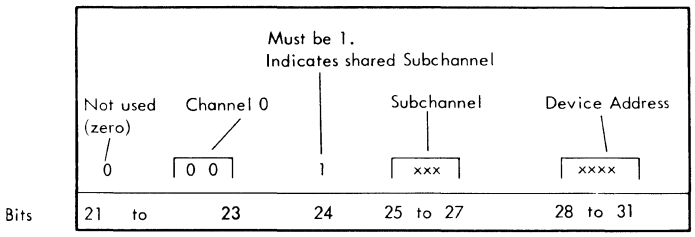


Figure 3B-19. Shared Subchannel Address Format

Start I/O Execution

Start I/O execution begins at the time that the common channel decodes a channel 0 operation and consists of five phases:

Channel and subchannel interrogation.
CAW-fetch.
CCW-fetch.
Initial selection.
Store channel.

The start I/O execution is complete, as far as the CPU is affected, as soon as a condition code is set. The condition codes which may be set during start I/O are:

Condition code 00; indicates that the start I/O has been successfully started.
Condition code 01; indicates that the status portion of the CSW has been stored.
Condition code 10; indicates that either the channel is operating in burst mode or that the selected subchannel is busy.
Condition code 11; indicates that the channel, subchannel or device is not operational.

The setting of any condition code causes 'end execute' to be set and the CPU to proceed with the next sequential operation.

Channel and Subchannel Interrogation

- The common channel decodes the channel address.
- The channel and subchannel are checked for busy conditions.
- The device address is checked for validity.

At the end of I-fetch, the compute clock is started and, at CC6, the channel decode latches are set in the common channel.

If B register bits 22 and 23 are zero, a channel 0 operation is decoded. If an invalid channel address is found, a condition code of 11 is set and the operation is terminated.

The 'channel 0 burst mode' latch is now tested. If it is on, a condition code of 10 is set and the operation is terminated.

If the channel is not operating in burst mode, the 'data flow occupied' latch is tested. This latch, if on, signifies that the channel is at present handling a byte of information; therefore the start I/O operation must wait until the data flow is free. If the channel remains busy for more than approximately 100 microseconds, the 'burst mode' latch is set and condition code 10 is set as before.

The 'I/O execute' latch is now set and the interface clock is started. The contents of B register bits 24 to 31 are set into the CAR and into the 'generate extension storage address' circuits. This places the address of UCW 1 into the SAR.

The 'UCW fetch request' latch is turned on and requests an I/O cycle. At RC4, CP2 of the main storage read/write cycle, the 'UCW fetch cycle' latch is set which causes the contents of UCW 1 to be set into the channel registers. A second I/O cycle is now taken. The address of UCW 2 is generated by forcing bit 29 of the 'address in' funnel and the contents of UCW 2 are set into the channel.

During this cycle, a check is made of the 'end' flag and command register contents which were set during the previous cycle. If any bits are present, it signifies that the selected subchannel is already operating; therefore, a condition code of 10 is set and the operation is terminated.

CAW-Fetch

- CAW is read out of main storage location 48 hex.
- CCW address is loaded into D and E registers.

Before the I/O operation can commence, it is necessary to load the contents of the CCW into the channel. The CCW may be located anywhere in main storage but its address is in the CAW. The CAW format is shown in Figure 3B-20.

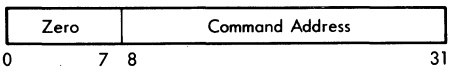


Figure 3B-20. CAW Format

At WC2, CP2 of the second UCW cycle, the 'channel address word' latch is set if the selected subchannel is free. At the same time, the 'control word fetch request' latch is turned on and the combination of CAW latch and 'control word fetch request' requests a C-cycle. Also at the same time, the address of the CAW (48 hex) is generated and set into the SAR.

The CAW bits 14 to 31 are set into the D and E registers. Because the maximum storage size of the IBM 2044 is 128K, ones in bit positions 8 to 14 signify an invalid address; bits 8 to 14 are, therefore, compressed into bit 14. A one is now set into bit 27 of the D register, thus increasing the value held by eight and generating the address of the next sequential CCW. At this point a program check is set if the CAW is not of the correct format, that is, if bits 0 to 7 and 29 to 31 are not zero.

An I/O cycle is now taken, during which the updated contents of the D register are stored into UCW 3. This is done by setting the 'UCW store request' and 'word 3 request' latches. Word 3 request sets a one into bit position 28 of the extension storage address, thus generating the address of UCW 3. The D register is then gated to the SDR. The address of the next CCW is now contained in UCW 3.

CCW-Fetch

- CCW is fetched from main storage and set into the channel.
- Two I/O cycles are required.

The CCW may be located anywhere in main storage. The address is contained in the E register after the CAW-fetch cycle. The format of the CCW is as shown in Figure 3B-21.

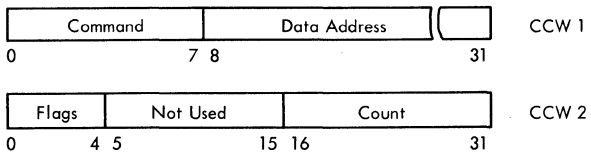


Figure 3B-21. CCW Format

During the I/O cycle that is used to store UCW 3, the 'control word fetch request' latch is turned on at WC2, CP2. This latch requests an I/O cycle and gates the contents of the E register through the address funnel into the SAR; CCW 1 is thereby fetched and its contents are set into the command register (bits 2 to 7) and the D register (bits 14 to 31). Bits 2 to 7 are also set into the CDR for use during the initial-selection sequence. During this cycle, the format of CCW 1 is checked and a program check is set if any of the following conditions occur:

Bits 4 to 7 equal to zero.
Bits 4 to 7 signify a Transfer-in-Channel (TIC).
Invalid address.

The 'control word fetch request' latch is set once more to initiate the second I/O cycle during which CCW 2 is set into the channel. In order to generate the CCW 2 address, the E register is again gated to the SAR; on this occasion, bit 29 is forced. The forced bit effectively increases the address by four and creates the address of CCW 2. CCW 2 is now read out of main storage and its contents are set into the channel registers as follows:

Bits 0 to 4 into the flags register.

Bits 16 to 31 into the E register, forming the byte count.

The contents of both CCW 1 and CCW 2 are now contained in the channel registers and the device can be selected.

Initial Selection

- Initial selection is the sequence used to select the control unit and device from the channel.
- The control unit and device address is sent to interface bus out.
- The control unit responds with its address on bus in.
- The command is sent to bus out.
- The control unit sends a status byte to bus in.
- The status byte is analyzed.

The channel is connected to the control units by means of the standard interface, a full description of which is contained in IBM System/360 I/O Interface-Channel to Control Unit (OEMI), Form A22-6843.

During the initial-selection sequence, no main-storage cycles are required; the movement of data within the channel is controlled by the interface clock. A description of the interface clock is contained under "System Control Components" in Chapter 2 of Introduction and Functional Units, Form Y33-0001.

The initial selection is started during the second CCW cycle. At WC4, the 'first cycle' latch and the 'start selection' latch are turned on. The start selection latch starts an interface-clock cycle and, at clock pulse SA, the 'address out' latch is turned on. This latch gates the address contained in the CAR to the interface bus out lines. The 'address out' tag is also generated. Each control unit now attempts to decode the address byte. At the end of the interface-clock cycle, the signals 'hold out' and 'select out' are produced. When the selected control unit receives the 'select out' tag, it responds by placing its address on bus in and raising the 'address in' tag.

The 'address in' tag starts another interface-clock cycle. At clock pulse S3, the address on bus in is compared with the content of the CAR to ensure that the correct control unit has been selected. A mismatch sets the interface check. Also at S3, the 'command out' latch is set, which gates the command from the channel data register to bus out and so to the control unit. The 'command out' tag is now raised.

When the control unit has decoded the command, it places a status byte on bus in and brings up the 'status in' tag. This starts a further interface-clock cycle during which the status byte is gated into the channel data register and analyzed. At interface clock pulse S3, the 'service out' tag is sent to the control unit.

If the status is zero, a condition code of 00 is set into the common channel and 'end execute' is turned on. The CPU is now released and the channel stores UCW's 1 and 2. If, however, a status other than zero is received, the status portion of the CSW is stored and the operation is terminated (see "CSW Storing") under "I/O Interrupts."

Store Channel

- Two I/O cycles are taken to store UCW's 1 and 2.

When the control unit receives the 'service out' tag, it responds by dropping 'operational in'. This action produces the 'burst mode ended' signal which turns on the 'UCW store request' latch to request an I/O cycle. The address contained in the CAR is gated to the extension-storage-address-generating circuits, the address of UCW 1 then being gated to the SAR.

The contents of the command register and D register are stored in the UCW 1 location. During this I/O cycle, a second I/O cycle is requested, bit 29 is forced in the address funnel, thus generating the address of UCW 2. The contents of the flags register, channel status register and E (count) register are stored in UCW 2.

The start I/O operation is now complete; the UCW's contain all the information required to perform the channel operation and the control unit has received the command.

Data Service

- Data service is initiated by the control unit.
- Channel 0 can perform read forward, read backward or write operations.
- One byte is transferred between the channel and SDR during each data service.

Flow charts and timing charts for the data service operation are shown in FEMD Figures 6105 to 6108.

The data service operation is always started by the control unit. When the control unit has a byte of data to be transferred to the CPU, or requires a byte of data from the CPU, it signals the channel by raising the 'request in' tag.

It is possible that more than one control unit may require service at the same time. If this occurs, the priority is decided by the way in which the select lines pass through the control units. In each control unit, the 'select out' and 'select in' lines may be jumpered either through the internal selection circuits of that unit or directly to the next control unit. Figure 3B-22 illustrates two examples of jumpering.

The first control unit on the standard interface can be connected so that it has either the highest or the lowest priority. Every other control unit can have any priority.

During each data service, the command register is used to decide the direction of data transfer (input or output). If command register bit 7 is zero, the data flow is from the device to main storage (read, read backward or sense); if the bit is a one, the data flow is from main storage to the device (write or control).

Each data service therefore results in one byte of data being transferred either to or from the device.

Request-In for Service

- The control unit sends 'request in' to the channel.
- UCW's 1 and 2 are loaded into the channel registers.
- 'Command out' tag is sent to the control unit when the channel is ready to continue the operation.

The 'request in' tag from the control unit can initiate the data service sequence only if the channel is not already operating. The channel is free if all the following latches are off:

Data flow occupied
I/O execute
Select interrupt.

If these latches are off, the 'request in' signal sets the 'device request' latch, the 'select out' tag then being sent to the control unit. The control unit then places the device address on bus in and raises the 'address in' tag, which starts an interface clock cycle in the channel. The 'address in' tag also sets the 'data flow occupied' latch.

At interface clock pulse SA, the device address on bus in is set into the CAR and, at pulse S2, the 'UCW-fetch request' latch is set on. An I/O cycle is requested and the address contained in the CAR is

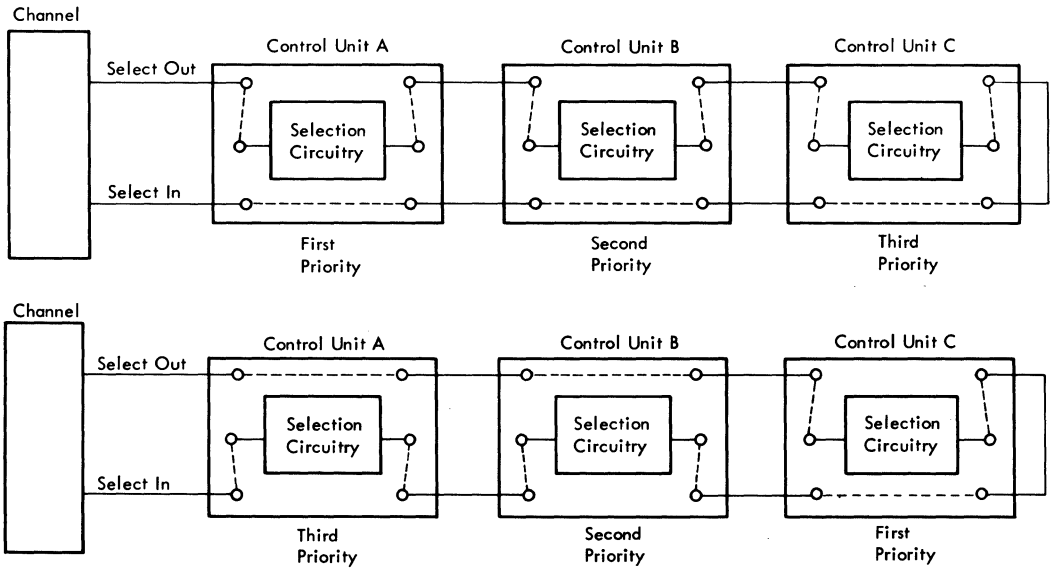


Figure 3B-22. Interface Priority

gated to the SAR through the generate-extension-storage-address circuits. At RC4, CP2, the 'UCW-fetch cycle' latch is turned on. The contents of UCW 1 are now in the SDR. At WC2, CP2, bits 14 to 31 (data address) are set into the D register and, at WC3, CP1, bits 2 to 7 are set into the command register.

At WC2, CP2, the 'UCW-fetch request' latch is again set on and a second I/O cycle is requested. During this cycle, the address sent to the SAR is modified by the insertion of a one into bit position 29, thus producing the address of UCW 2. At WC3, CP1 of this cycle, bits 0 to 4 are set into the flags register, bits 8 to 14 into the channel status register and bits 8 to 31 (bits 8 to 14 compressed into bit 14) into the E register.

The channel now contains all the information required to execute the data transfer to or from the device. The 'command out' tag is raised to signal the control unit to proceed.

Service In

Read Forward

When the control unit receives the 'command out' tag for a read operation, it sets the byte of data to the bus in lines and raises the 'service in' tag. This starts an interface-clock cycle and, at clock pulse SA, the byte of data is set into the channel data register.

If the E register byte count is not zero and the 'halt' flag is not set, the 'data request' latch is turned on at pulse S2. This latch requests an I/O cycle, during which the D register contents are set into the SAR. The contents of the channel data register are gated through the data funnel into the SDR, the byte position being decided by the D register bits 30 and 31.

At RC2, CP2, the byte count in the E register is reduced by one and, at WC4, CP2, the data address in the D register is increased by one.

If the E register now contains zero, the CD flag is tested and, if it is on, the data chain control latch is set.

At RC4 of the I/O cycle, the 'service out' latch is set and the 'service out' tag is sent to the control unit to indicate that the data byte has been accepted. If the control unit does not have another byte of data to be transferred, it drops the 'operational in' tag and UCW's 1 and 2 are stored. This operation is described previously under the heading "Store Channel."

Read Backward

The read-backward operation is identical to a read-forward operation, except that the data address in the D register is reduced by one during the data cycle instead of being increased.

Write

During a write operation, data is passed from main storage to the control unit. The write operation proceeds in the same way as the read operation up to the point when the control unit sends 'service in' to the channel. During the resulting interface-clock cycle, no data is set into the channel data register from the interface bus in but, at WC2 of the data cycle, the contents of the SDR are gated into the channel data register. The correct byte is selected according to the contents of bits 30 and 31 of the D register. The channel data register is then gated to the interface bus out and the 'service out' tag is raised. As with the read operation, the value held in the E register is reduced by one and the value held in the D register is increased by one.

The two UCW store cycles are then initiated.

Ending Procedure

- Either the channel or the control unit can recognize the end of a data transfer.
- 'Channel end' is generated by the control unit when the transfer of data is complete.
- 'Device end' is generated when the device reaches its normal ending point.
- Both 'channel end' and 'device end' cause an interrupt request.

The flow chart and timing chart for the channel-end and device-end sequences are contained in FEMD Figures 6113/6114.

It is possible for either the channel or the control unit to recognize the completion of a data transfer. The channel does so when the E register (byte count) goes to zero. The way in which the control unit recognizes the end depends on the type of device and the operation being carried out. In the case of a channel-initiated end procedure, the channel signals the control unit that it cannot take any further data bytes. The control unit then presents a status byte to the channel as soon as it can do so. If the control unit recognizes the end, it sends a status byte to the channel without any signal from the channel.

The status byte from the control unit contains the 'channel end' bit (bit 4) if the data transfer is complete and the 'device end' bit (bit 5) if the device has reached its normal ending point. The status byte is shown in Figure 3B-23.

Bit	Description
0	Attention
1	Status Modifier
2	Control Unit End
3	Busy
4	Channel End
5	Device End
6	Unit Check
7	Unit Exception

Figure 3B-23. Status Byte

When the channel receives a 'channel end' from the control unit it sets the 'end' flag in the command register and stores the status byte in UCW 2. An I/O interrupt is then requested. If a status byte from another control unit is received before the interrupt is taken, the byte is stacked back into the control unit until the channel is free to accept it.

When the channel receives a 'device end' status byte, it always stacks the byte back to the control unit.

'Channel end' and 'device end' may occur either at the same time or independently. If they occur at the same time, only one interrupt is requested during which they are both processed.

Channel-Initiated End Procedure

- The channel recognizes a byte count of zero during a 'service in'.
- A 'command out' of zero is sent back to the control unit.

During a data service it is possible that the byte count is zero. The channel tests for this condition during the 'service in' interface-clock cycle. If the E register byte count is zero and the CD flag is not on, the line 'stop or stack' sets the 'command out' latch. No bits are set to the interface bus out. The control unit recognizes this condition as a 'channel end'. As soon as possible it presents the status byte to the channel during a request-in-for-status sequence.

Request-in for Status

- The control unit sends 'request in' to the channel.
- The UCW's are loaded into the channel.
- The status byte is stored in the channel data register.

The channel commences a normal request-in sequence when it receives the 'request in' from the control unit. The channel sends 'select out' to the control unit which responds with its address on bus in and an 'address in' tag. The two UCW-fetch cycles then proceed in the same way as during a request-in-for-service sequence. When the UCW's are loaded, the channel sends 'command out' to the control unit. The control unit then sets the status byte to the bus in lines and raises the 'status in' tag. This starts an interface-clock cycle during which, at clock pulse SA, the status byte is set into the chan-channel data register.

Status In (Channel End)

- If there are no interrupts pending, the interrupt buffer register is set from the CAR and an interrupt is requested.
- If there are any interrupts pending, the status is stacked back to the control unit.

At S2 of the interface-clock cycle that is started by the 'status in' tag, the 'end status' latch is set from bit 4 of the channel data register. This latch sets the 'end' flag in the command register. At S3, 'service out' is sent to the control unit.

If the interrupt request latch is off, at S4 the 'channel end interrupt request' and 'active end interrupt request' latches are both set and the command register bits 4 to 7 are reset. If the interrupt request latch is on, the status is stacked back to the device. The sequence used to accomplish this stacking is described subsequently in "Status In (Device End)."

When 'operational in' falls, the UCW store request latch is turned on, which requests two I/O cycles. During the first I/O cycle, the command and channel address registers are loaded into UCW 1 and the second cycle is requested. During the second cycle, the contents of the channel data register (status byte), the channel status register and the E register are loaded into UCW 2. The contents of UCW 1 and 2 are now as shown in Figure 3B-24.

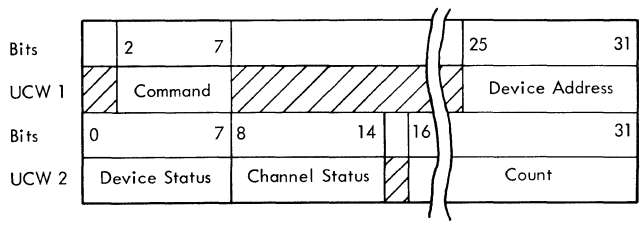


Figure 3B-24. UCW 1 and 2 Contents after Channel End

At WC3 of this second I/O cycle, the CAR contents are transferred to the interrupt buffer register. At WC4, CP2, the 'active end interrupt' latch is turned off and the 'I/O interrupt' latch is turned on. The channel now waits until the CPU is able to accept the interrupt.

Status In (Device End)

- If device-end status occurs before the channel-end interrupt has been taken, the status is stacked back.

- If channel-end interrupt has been taken and no other channel 0 interrupts are pending, the interrupt buffer register is loaded and the status stacked back.

Two conditions may exist in the channel when a device-end interrupt is received. The channel-end interrupt may still be pending; in this case, the only action is for the channel to stack the status back to the control unit. If, however, the channel-end interrupt has already been taken, the 'device end interrupt' latch is set and the device address is loaded into the interrupt buffer register.

Device-end status is presented to the channel during a normal request-in-for-status operation. When the channel receives the 'status in' tag, an interface-clock cycle is started. At clock pulse SA, the status byte is loaded into the channel data register. The 'device end interrupt request' latch is set if, at S2 of this cycle, the following conditions are present:

No 'channel end' bit in the status byte
No 'end' flag in the command register
No I/O interrupts pending
Not first cycle.

At the same time as the latch is set, the device address is transferred from the CAR to the interrupt buffer register. The line 'stack status' is now brought up, and 'command out' is sent to the control unit. A 'command out' response to 'status in' signifies to the control unit that the status byte is to be stacked. The 'suppress out' interface line is now brought up and remains up until the interrupt request is accepted by the CPU.

Two UCW store cycles are now initiated as for a 'channel end'. During the UCW 1 store cycle, the command register and the D register are loaded into UCW 1; UCW 1, however, has been previously reset. During the second cycle, the flags register, status register and E register are loaded into UCW 2. The UCW contents are now as shown in Figure 3B-25.

If the channel end interrupt is still pending when 'status in' is presented to the channel, the status is stacked back to the control unit as previously described. The interrupt buffer register is not loaded with the device address and no UCW store cycles are taken. If any other interrupt is pending, the status is stacked back and the two UCW store cycles are taken.

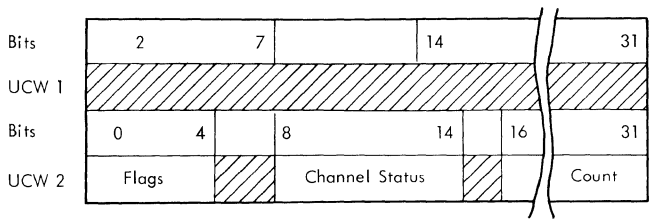


Figure 3B-25. UCW 1 and 2 Contents after Device End

CHAINING

- Chaining allows the execution of a new CCW without requiring a start I/O operation.

- Two types of chaining are provided: data chaining and command chaining.

- Transfer-in-channel command allows CCW chains to be arranged in non-contiguous storage areas.

Chaining allows more than one CCW to be executed without requiring a start I/O instruction for each CCW. Two flags are provided in the CCW to cause chaining to take place, the CD flag for data chaining and the CC flag for command chaining. Normally, the CCW's are arranged in storage in successive double-word locations. The address of the next CCW can be found by adding eight to the address of the present CCW. Two chains located in non-contiguous storage areas can be coupled by a transfer-in-channel command.

Further information on chaining operations is contained in IBM System/360 Principles of Operation, Form A22-6821.

Data Chaining

Summary of Operation

- The CD flag causes a data chaining operation to be started as soon as the byte count goes to zero.

- A new CCW is fetched from main storage.

The flow chart and timing chart for data chaining are shown in FEMD Figures 6109/6110.

During the data-chaining sequence, a new CCW is fetched from main storage and loaded into the subchannel.

Initiation of Data Chaining

- Data chaining is initiated when the byte count goes to zero.

The data-chaining operation is initiated during a request-in sequence if the byte count goes to zero. At WC2, CP2 of the data cycle, the 'data chain control' latch is turned on. The conditions required to set this latch are:

Data cycle
E register byte count zero
CD flag.

CCW-Fetch

- The address of the next CCW is fetched from UCW 3.
- The new CCW is loaded into the channel registers.
- The subchannel is loaded with the new CCW date.

The conditions used to set the 'data chain control' latch also produce the line 'chaining request UCW 3 update'. This line sets the 'UCW-fetch request' latch and the 'word 3 request' latch. 'UCW-fetch request' requests an I/O cycle during which the address contained in the CAR is gated into the SAR. A one is forced into bit position 28, generating the address of UCW 3 for that subchannel.

At WC2, CP2, the contents of UCW 3 (next CCW address) are gated into the D and E registers. The value held in the D register is then increased by eight to produce the address of the next CCW. The 'UCW store request' and 'word 3 request' latches are then set on and another I/O cycle is requested. During this cycle, the content of the D register (next CCW address) is set into the SDR and loaded into UCW 3.

The 'control word fetch request' latch is turned on at WC2, CP2 of the UCW 3 store cycle. This requests a further I/O cycle, during which the E register content (CCW address) is gated to the SAR. Bits 14 to 31 (data address) of the CCW 1 are gated into the D register. Bits 4 to 7 (command) are not set into the command register as this is a data-chaining operation.

A second CCW-fetch cycle is now initiated. The E register is again gated to the SAR, but a one is forced into position 29 of the SAR which creates the address of the second word of the CCW (CCW 2). During this cycle, the contents of CCW 2 are set into the E register and the flags register.

The new CCW contents have now been loaded into the channel registers. If the channel is operating in burst mode, or 'operational in' has not dropped, the next data service is handled as normal.

If 'operational in' is not up, two UCW store cycles are taken to store the channel register contents into UCW's 1 and 2.

Command Chaining

Summary of Operation

- The CC flag causes a command-chaining operation to be started when 'device end' is received.
- A new CCW is fetched from main storage.
- The new command is sent to the control unit.

The flow chart and timing chart for command chaining are shown in FEMD Figures 6111/6112.

During the command-chaining sequence, a new CCW is fetched from main storage and set into the channel; the control unit is then reselected and the new command is issued.

Command chaining takes place only if no unusual conditions have been detected in the current operation. If any of the following conditions arise, command chaining is suppressed and an interrupt is requested:

Unit check
Unit exception
Incorrect length with no SLI flag
Program check.

Initiation of Command Chaining

- Command chaining is initiated when device-end status is received from the control unit.

The setting of the 'command chain control' latch is shown in Figure 3B-26. In the case of an immediate operation, the control unit sends channel and device-end status during the initial-selection sequence. At this time, the byte count will not be zero. This condition is detected by the 'first cycle' latch being on when 'device end' (status bit 5) is recieved.

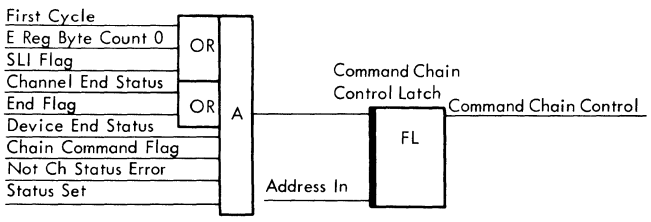


Figure 3B-26. Command Chain Control Latch

In all other cases, the command chain control latch is set only if either the E register byte count is zero or an SLI flag is present.

CCW-Fetch

- The address of the next CCW is fetched from UCW 3.
- The new CCW is loaded into the channel registers.
- Initial-selection sequence is started.

The fetching of the CCW's for a command-chaining operation is accomplished in the same way as for data chaining. However, during the CCW 2 cycle, the command register is set with the new command. The command byte is also set into the channel data register for use during the initial-selection sequence.

Initial Selection

- The control unit and device are reselected.
- The new command is sent from the channel data register to the control unit.

At WC4 of the second CCW-fetch cycle, the combination of 'command chain control' latch and 'control word fetch word 2' sets the 'start selection' latch. The initial-selection sequence is then executed as described previously in "Start I/O Operation."

If, at the end of this sequence 'operational in' is not active, two UCW store cycles are taken to store the new CCW data into the selected subchannel.

Transfer-In-Channel

Summary of Operation

- Transfer-in-channel is used to provide chaining between CCW's not located in contiguous storage locations.

The TIC command provides a connection between two chains of CCW's or breaks a chain of sequential CCW's. It is, in effect, a "branch" instruction. When the channel recognizes a TIC-CCW, the new CCW is fetched from main storage and loaded into the channel. The format of a TIC-CCW is shown in Figure 3B-27.

The flow chart and timing chart for the TIC operation are shown in FEMD Figures 6111/6112, while the sequence is shown in Figure 3B-28.

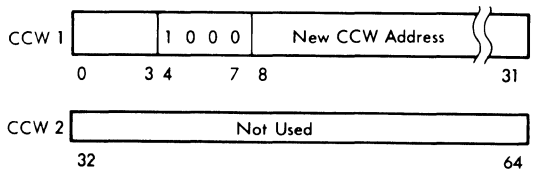


Figure 3B-27. Transfer-in-Channel, CCW Format

Recognition of TIC

- A TIC is recognized by the channel when a CCW is fetched furing a chaining operation.

The TIC command is recognized by the channel during the third I/O cycle of a chaining operation. During this cycle, the E register contents are used to address the new CCW in main storage.

When the new CCW is in the SDR, bits 4 to 7 (command code) are tested for a TIC command (1000). If this is present, the 'TIC control' latch is set. A test is then made of the following conditions:

Start I/O
CAW-cycle
TIC control on from last CCW
SDR bits 29 to 31 not zero
Valid CCW address in SDR bits 8 to 16.

If any of these conditions are found, a program check is set and an interrupt is requested. Bits 14 to 31 of the CCW (next CCW address) are set into the D and E registers. The value held in the D register is then increased by eight to produce the address of the next CCW.

At WC3, CP1 of this third chaining cycle, the 'UCW store request' and 'word 3 request' latches are turned on and a further I/O cycle is requested. The address of UCW 3 is gated into the SAR and the contents of the D register are stored in UCW 3. The address of the new CCW plus eight is thus stored in UCW 3.

Branch to New CCW

- The address contained in the E register is used to fetch the new CCW.

At WC2, CP2 of the UCW 3 store cycle, the 'control word fetch request' latch is turned on and a main storage cycle is requested.

The E register contents are gated to the SAR and the new CCW is fetched and loaded into the channel registers. The set of the command register is controlled by the CC flag.

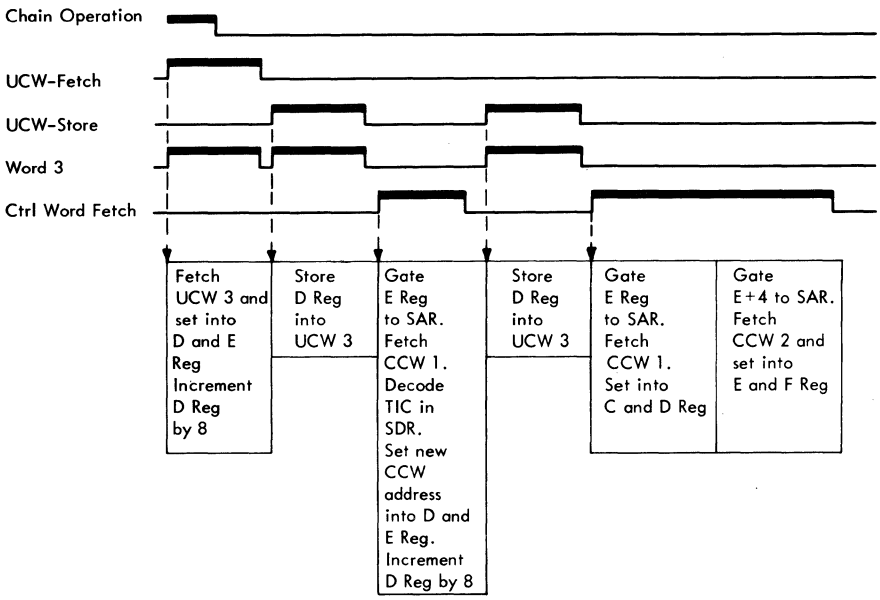


Figure 3B-28. Transfer-in-Channel Sequence

The operation then continues as a normal chaining operation; another CCW-fetch cycle is initiated and the contents of CCW 2 are gated into the channel registers.

If the TIC was encountered during command and chaining the device is reselected. If it was encountered during data chaining, the channel registers are stored into UCW's 1 and 2.

Status Modifier

Purpose of Status Modifier Bit

- Status modifier bit causes the normal sequence of commands to be altered.

The status modifier bit is bit 1 of the status byte. If it is presented by the device with a 'device end' bit during a command-chaining operation, the normal sequence of commands is altered. The presence of the status modifier bit causes the new CCW to be taken from an address 16 bytes higher than the current CCW.

NOTE: A new CCW is normally eight bytes above the current CCW.

An example is given of the use of the status modifier bit during search operations on disk files; the sequence of CCW's for this operation is shown in Figure 3B-29. The operation is started by issuing to the disk file a CCW that specifies a search operation. If, at the completion of this CCW, no search-equal condition has been found, the TIC-CCW is executed which causes the search CCW (CCW A) to be executed again. This continues until either a search-equal or a search-end condition is found.

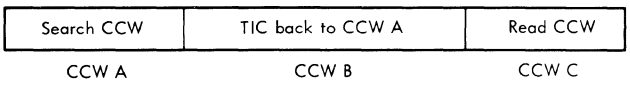


Figure 3B-29. CCW Chain for File Search

If a search-equal condition is found, the file control unit presents a status modifier bit to the control unit when CCW A is completed. This causes the channel to generate a new CCW address 16 bytes higher than CCW A, thus by-passing the TIC command (CCW B). The read CCW (CCW C) is then executed to transfer the required record to main storage.

Increment CCW Address by 16

The status modifier control latch is set when the device presents device-end status if a CC flag is present in the channel. At WC2, CP2 of the UCW 3 store cycle, if the status modifier control latch is on, the line 'chaining request UCW 3 update' is again brought up. This causes UCW 3 to be fetched again and its contents to be set into the D and E registers. The value held in the D register is increased by eight and a UCW 3 store cycle is requested. The D register contents are now the current CCW address plus 24 bytes and the E register contains the current CCW address plus 16. The command-chaining sequence then continues normally.

TEST I/O OPERATION

Summary of Operation

- Test I/O is used to find the current state of a channel, subchannel and device.
- Under certain circumstances a pending interrupt may be reset.
- Result of the operation is shown in the condition code and, under certain circumstances, by storing the CSW.

The instruction 'test I/O' is used by the program to establish the current condition of the addressed channel, subchannel and device. The result of the operation is sent to the CPU in the condition code. If the channel and subchannel are free, a status byte is fetched from the control unit, the CSW is stored and any pending interrupt for that device is cancelled.

In this case it is possible to cancel the interrupt because the test I/O instruction has performed the function of the interrupt for that device.

The condition codes that may be set by a test I/O instruction are as follows:

00 : Device was available.
01 : CSW was stored.
10 : Channel or subchannel was busy.
11 : Channel, control unit or device was not operational.

Flow charts and timing charts for this operation are contained in FEMD Figures 6115 to 6118, while Figure 3B-30 gives a simplified flow.

Test I/O Execution

I-Fetch

- The instruction 'test I/O' is decoded by the CPU.
- The channel, subchannel and device address is generated in the B register.

The instruction 'test I/O' uses the SI format. The CPU must be in the supervisor state when the instruction is executed.

A description of I-fetch is contained in Principles of Operation - Processing Unit, Form Y33-0002. At the end of I-fetch, the address of the channel, subchannel and device is contained in the B register bits 21 to 31.

Channel and Subchannel Interrogation

- The common channel decodes the channel address.
- The channel and subchannel are tested for busy conditions.
- The setting of any condition code causes 'end execute' to be generated.

At the end of I-fetch, the compute clock is started and, at CC6, the channel decode latches are set in the common channel. If B register bits 22 and 23 are zero, a channel 0 operation is decoded. If an invalid channel address is decoded, a condition code of 11 is set and the operation is terminated.

The 'channel 0 burst mode' latch is now tested. If it is on, a condition code of 10 is set to signify that the channel is busy. If the channel is not working in burst mode, but the 'data flow occupied' latch is on, then the test I/O operation waits until either the 'channel 0 burst mode' latch is set or the 'data flow occupied' latch is turned off.

The 'I/O execute' latch is now set and the interface clock is started. B register bits 24 to 31 are set into the CAR and also to the generate-extension-storage-address circuitry.

The address of UCW 1 for the selected subchannel is gated to the SAR and the 'UCW fetch request' latch is set. Two I/O cycles are taken, during which the channel registers are loaded from UCW's 1 and 2.

During the UCW 1 cycle, the device address contained in the CAR is compared with bits 25 to 31 of UCW 1. If a 'channel end' is pending, UCW bits 25 to 31 contain the address of the device that presented the 'channel end'. The result of this comparison is stored in the 'match data out' latch for use during the UCW 2 cycle.

During the UCW 2 cycle, the contents of UCW 1 are analyzed to find the condition of the addressed subchannel. At this time, three conditions may cause a condition code of 10 to be set, indicating that the subchannel is busy. These conditions are:

1. No 'end' flag ('channel end' not received) and command register not zero. This signifies that the addressed subchannel is working in byte mode.
2. 'End' flag and 'command chain control' both on. This signifies that the channel is waiting for a 'device end' before continuing a chaining operation.
3. 'End' flag on and 'match data out' latch off. This signifies that the pending 'channel end' is for a device other than the one to which the 'test I/O' is addressed but sharing the same subchannel.

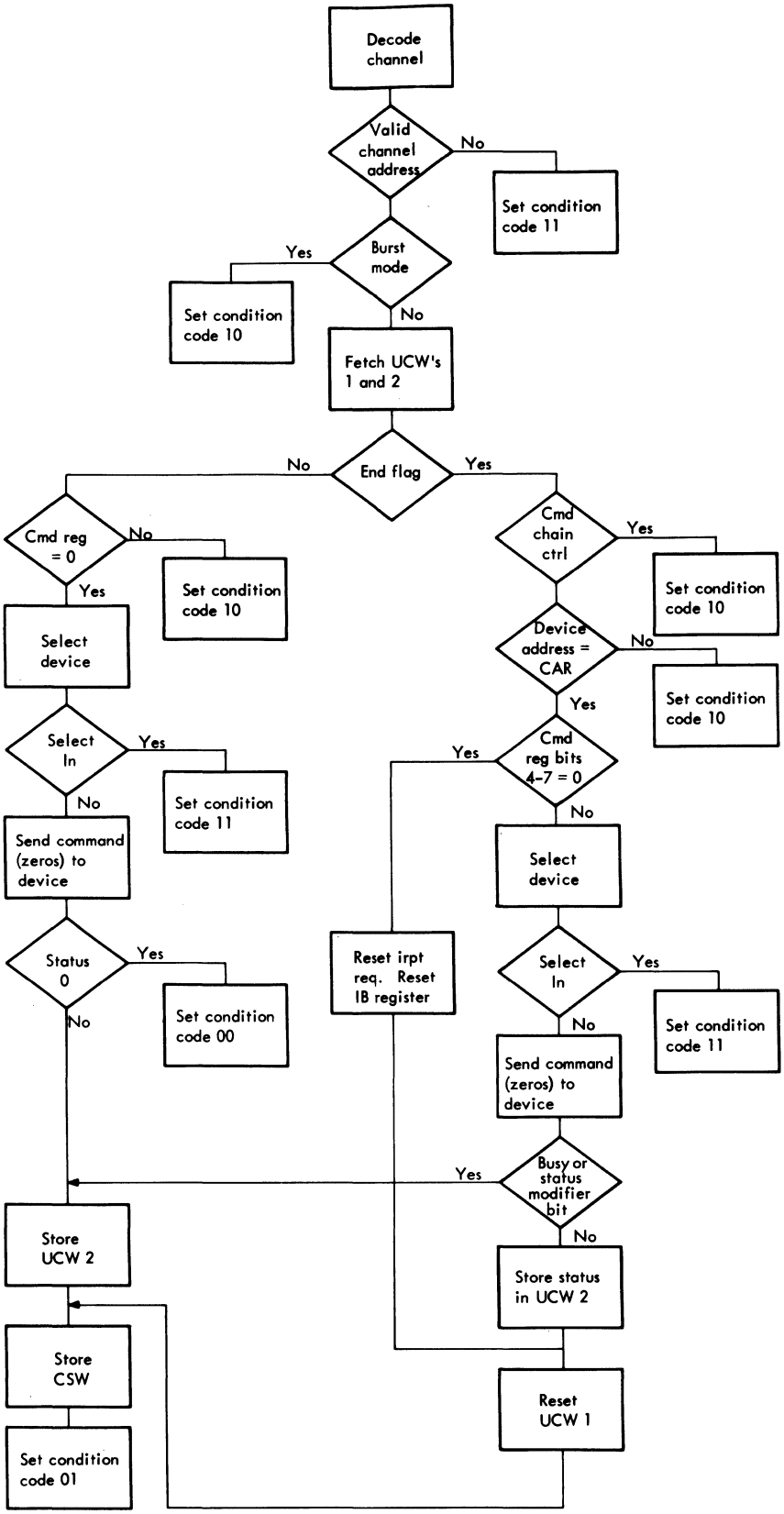


Figure 3B-30. Test I/O

If the 'end' flag is on and the command register bits 4 to 7 are zero, the channel has accepted the status byte from the device and an interrupt is pending. In this case the test I/O operation is used to store the CSW and the interrupt request is cancelled. During the UCW 2 cycle, the line 'IB register match' brings up last interrupt execute which resets the interrupt buffer (IB) register and the 'interrupt request' latch.

Under these circumstances, the address in the interrupt buffer register and the CAR must be the same because the CAR has previously been compared with the device address contained in UCW 1.

The 'UCW 1 reset cycle request' latch is set and an I/O cycle is taken during which UCW 1 is reset. The CSW is then stored as described subsequently in "I/O Interrupts" under the heading "CSW Storing."

For each of the test I/O conditons so far described, it was possible for the status to be decided by examination of the channel and subchannel; for the remaining conditions it is necessary to reselect the device and obtain a status byte.

Initial Selection

- The device is selected.
- The command is sent to the control unit.
- A status byte is presented to the channel by the control unit.

The initial-selection sequence is described previously in the "Start I/O Operation." The channel places the device address on bus out and the 'select out' tag is generated. If the response to 'select out' is 'select in', the device is not connected to the interface and a condition code of 11 is set. The response of the control unit to 'select out' is either 'address in' if it is free, or 'status in' if it is busy. If the response is 'address in', the channel sends the 'test I/O' command byte (all zeros) to the control unit with a 'command out' tag; the control unit then sets the status byte to bus in with a 'status in' tag. If the control unit is busy, the status byte is presented with the 'status in' tag when 'address out' is sent.

Status Analysis

The status byte is sent into the channel data register and analyzed, depending upon the condition of the 'end' flag. The five conditions that may exist, and the action, are as follows:

1. No 'end' flag, and status of zero. This condition signifies that the addressed device is free. A condition code of 00 is set and the operation is terminated.

2. No 'end' flag, and status not zero. In this case, the zero control latch is set and a UCW 2 store cycle is requested. This results in the status byte only being stored in UCW 2. Zeros are stored in the remaining bits of UCW 2. The CSW is then stored.

3. 'End' flag, and the status byte contains busy or status modifier bits. This condition signifies that either the control unit or the device is busy and an interrupt is pending for that device. In this case, the zero control latch is set and a UCW 2 store cycle is requested. The status byte is stored into UCW 2 and zeros are set to the remaining bits of UCW 2. The interrupt is not cleared.

4. 'End' flag, and the interrupt buffer register is equal to the CAR. This condition signifies that a PCI is pending for the addressed device. PCI status is set, and the interrupt buffer register and 'interrupt request' latch are reset. The device and channel status is stored into UCW 2. A UCW 1 reset cycle is taken to free the subchannel. The CSW is then stored.

5. 'End' flag, and the interrupt buffer register is not equal to the CAR. This condition indicates that the pending interrupt is not for the addressed device. In this case, the interrupt buffer register and 'interrupt request' latch are not reset. The device and channel status are stored into UCW 2. UCW 1 is then reset and the CSW is stored.

When the CSW is stored in the preceding cases, a condition code of 01 is set to indicate to the program that the result of the test I/O operation is stored in the CSW. A description of the CSW store cycles is included later in this manual in "I/O Interrupts."

Figure 3B-31 gives a summary of the condition code that is set for each condition, while the condition code generation circuits are shown in Figure 3B-32.

TEST CHANNEL

Summary of Operation

- 'Test channel' is used to find the current state of the addressed channel.
- The condition code is set to indicate the result of the operation.
- The subchannel and device are not interrogated.

The instruction 'test channel' is used by the program to establish the current condition of the addressed channel. At the end of the operation, the condition

Condition Code	General Interpretation	Comments
00	The channel, subchannel and device are all available.	The device has been selected and the status byte of zero sent to the channel during initial selection.
01	The CSW has been stored.	1. The subchannel was free but a bit was present in the status byte presented to the channel. 2. A pending interrupt has been cancelled and the status stored in the CSW. 3. The control unit has presented a status of busy and/or status modifier and an interrupt was pending. In this case, the interrupt has not been closed.
10	The channel or subchannel is busy.	1. The channel was operating in burst mode. 2. The selected subchannel contained a command and no 'end' flag. Equivalent to subchannel working. 3. The channel had a pending interrupt and the command chain control was active. 4. The channel has a pending interrupt for another device on the same subchannel as the one addressed by the 'test I/O' operation.
11	The channel, control unit or device is not operational.	1. An invalid channel address was decoded. 2. 'Select in' was received as a response to 'select out' during initial selection. The addressed device is not connected to the interface.

Figure 3B-31. Condition Code Interpretation for Test I/O

code is set to indicate the state of the channel. The interpretation of the condition code is as follows:

00 : Channel available
01 : Interrupt pending
10 : Channel operating in burst mode
11 : Channel not operational.

Test-Channel Execution

I-Fetch

- The instruction 'test channel' is decoded by the CPU.
- The channel address is set into the B register.

The instruction 'test channel' uses the SI format and is only executed when the CPU is in supervisor state. The I-fetch operation is described in Principles of Operation - Processing Unit, Form Y33-0002. At the end of I-fetch, the address of the channel is contained in the B register bits 21 to 23; 24 to 31 are ignored.

Channel Interrogation

- The channel address is decoded by the common channel.
- The condition code is set according to the condition of the channel.

At the end of I-fetch, the compute clock is started and at CC6 the channel address is decoded. An invalid address sets a condition code of 11. The 'burst mode' and 'I/O interrupt request' latches are then tested. If the 'burst mode' latch is active, a condition code of 10 is set and, if any interrupts are pending, a condition code of 01 results. If both the 'burst mode' and 'I/O interrupt request' latches are off, a condition code of 00 is set, indicating that the channel is available.

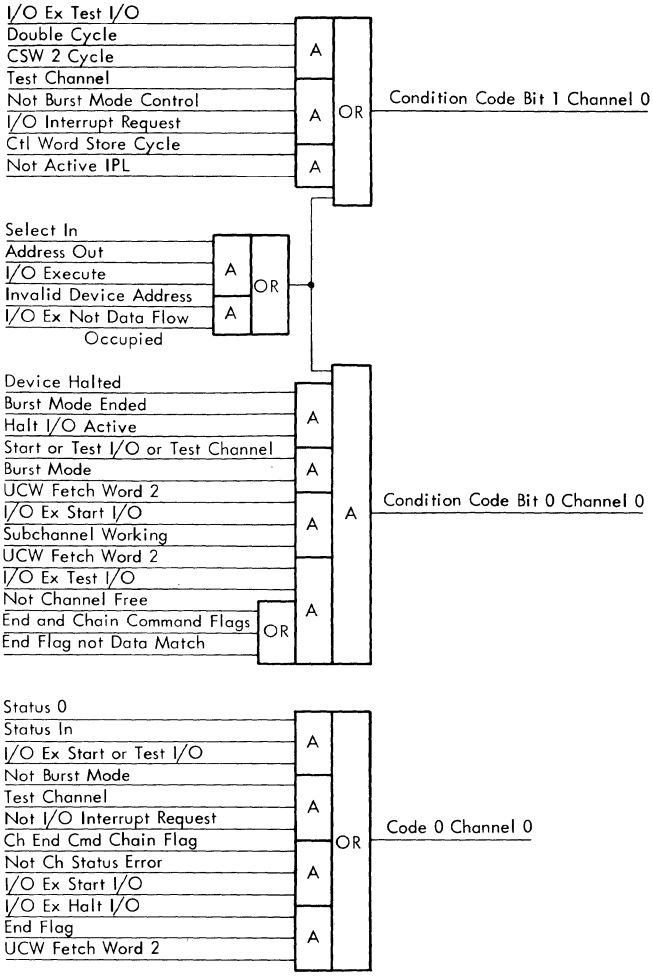


Figure 3B-32. Channel 0 Condition Code Generation

The condition code generation circuits are shown in Figure 3B-32. The setting of any condition code causes the 'end I/O execute' latch to be set and the operation to be terminated.

HALT I/O OPERATION

Summary of Operation

- Instruction 'halt I/O' is used to terminate the current I/O operation on the addressed device.
- If the channel is in burst mode, the burst operation is terminated.
- The condition code is set to show the result of the operation.

The instruction 'halt I/O' is used by the program to terminate an I/O operation. If the channel is not working in burst mode when the 'halt I/O' is issued, the addressed device is selected and its operation terminated. If the channel is in burst mode, the operation is terminated immediately, even though it may not be the addressed device that is operating. This action allows the program to clear the channel in order to start an operation of higher priority. If an interrupt is pending for the addressed subchannel, the 'halt I/O' has no effect.

At the end of the operation, the condition code is set to indicate the result of the operation. The condition codes that may be set and their meanings are:

00 : Interrupt pending in subchannel
01 : CSW stored
10 : Burst mode terminate
11 : Channel, subchannel or device not operational.

A flow chart and timing chart for 'halt I/O' is given in FEMD Figures 6119/6120.

Halt I/O Execution

I-Fetch

- The instruction 'halt I/O' is decoded by the CPU.
- The channel, subchannel and device address is generated in the B register.

The instruction 'halt I/O' uses the SI format and the CPU must be in the supervisor state when the operation is executed.

A description of I-fetch, is contained in Principles of Operation - Processing Unit, Form Y33-0002. At the end of I-fetch, the address of the channel, subchannel and device is contained in the B register bits 21 to 31.

Channel Interrogation

- The common channel decodes the channel address.
- The 'burst mode' latch is tested.
- The subchannel is tested for an 'end' flag.

At the end of I-fetch, the compute clock is started and at CC6 the common channel decodes the channel address. An invalid channel address causes a condition code of 11 to be set and the operation is terminated. The channel 0 'burst mode' latch is then tested. If it is on, the current burst-mode operation is terminated; if it is off, the control unit and device are selected.

Halt Burst-Mode Operation

- The chain flags are reset.
- An interface-disconnect sequence takes place.

If the channel is operating in burst mode, the line 'halt burst' becomes active at CC6. This resets the chain flags to stop the operation from being restarted when the device presents 'channel end' or 'device end', and sets the 'halt' flag in the command register.

The interface is disconnected by bringing up the 'address out' tag which signals the control unit to terminate the current operation. The device runs to a normal stopping point with no further data transfers.

When the control unit drops 'operational in', the 'device halted' latch is set and a condition code of 10 is set. 'End execute' is set to release the CPU.

The UCW store request latch is set by 'burst mode ended' when 'operational in' falls. This requests two UCW store cycles during which the command and D registers are stored in UCW 1 and the E flags and status registers in UCW 2, thus completing the operation.

Subchannel Interrogation

- The subchannel is checked for the presence of an 'end' flag.

At I/O CPA 1 (CPA and special CP1), if the channel is not operating in burst mode and the data flow occupied latch is off, the 'I/O execute' latch is set. If the 'data flow occupied' latch is on, the operation waits until either this latch goes off or the 'burst mode' latch comes on. The 'I/O execute' latch and interface-clock pulse S2 set the 'UCW fetch request' latch and two I/O cycles are taken to load the contents of UCW's 1 and 2 into the channel registers. During the UCW 2 cycle, the 'end' flag in the command register is tested. If the flag is on, the current operation for that subchannel is already complete and an interrupt is pending. A condition code of 00 is set and 'end I/O execute' releases the CPU. The two UCW store cycles are then taken as described previously in "Start I/O Operation."

If the 'end' flag is off, the device must be selected in order that the 'stop' command may be issued.

Device Selection

- The device is selected by an initial-selection sequence.
- A 'stop' instruction is issued if the control unit is not busy.

During the UCW 2 fetch cycle, if the 'end' flag is off, the 'start selection' latch is set and the 'halt' flag is turned on. The chain flags are also reset. 'Start selection' begins an initial-selection sequence (refer to "Start I/O Operation").

The sequence begins when the channel raises 'address out' and 'select out' to the interface with the device address on bus out. There are three possible responses from the control unit:

1. If 'select in' is received, a condition code of 11 is set and the operation is terminated.
2. If the control unit is free, it raises 'address in' to the channel which then drops the 'select out' tag and holds up the 'address out' tag.

NOTE: 'Address out' without 'select out' is, in effect, a 'stop' instruction.

3. If the control unit is busy, it responds with 'status in' and the status byte is set into the CDR.

When either 'address in' or 'status in' is received by the channel, the 'zero control' latch is set and a C-cycle is requested. When the C-cycle request is accepted by the CPU, the address of CSW 2 (44 hex) is generated and set into the SAR. The status byte is gated from the channel data register to byte 0 of the SDR. The channel status is not gated to the SDR as in a normal CSW 2 cycle because the 'zero control' latch is on.

At the end of the CSW store cycle, a condition code of 01 is set and 'end execute' releases the CPU. Two UCW store cycles are then requested during which the contents of the channel registers are loaded into UCW's 1 and 2.

INITIAL PROGRAM LOADING

Summary of Operation

- IPL is used to start program loading.
- The operation starts when the device address is set into the Load Unit Address (LUA) switches and the load pushbutton is pressed.
- Twenty-four bytes of data are read into main storage beginning at location 00.
- The double word read into location 08 is used as a CCW for the subsequent I/O operation.
- When 'channel end' is received, the channel stores the I/O address in bits 21 to 31 of main storage location 00 during a CSW store cycle.

Initial Program Loading (IPL) is the operation used to start the loading of a program into main storage. A summary of the operation is shown in Figure 3B-33. The operation is started when the operator

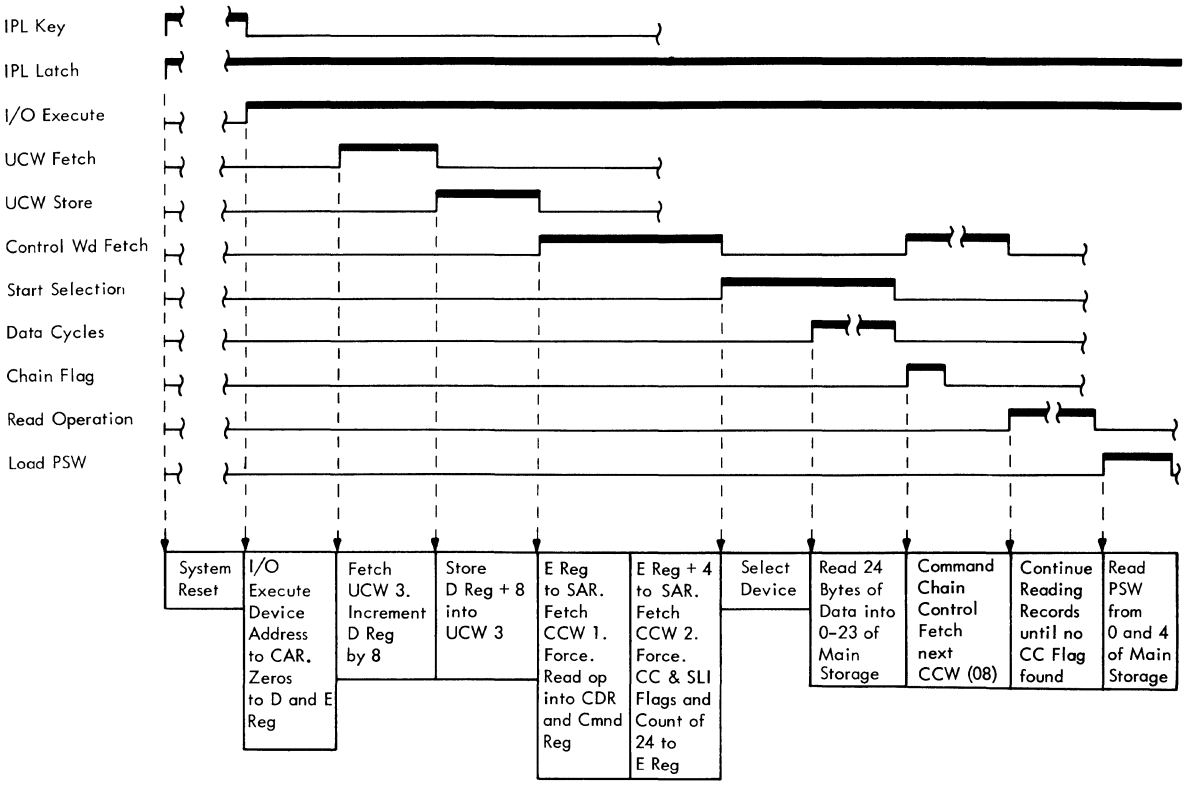


Figure 3B-33. IPL Operation

sets the LUA switches to the address of the required channel and device. The Load pushbutton is then pressed and a system reset occurs.

At the completion of the system reset, the channel is selected, a CCW is forced and a read operation is started. At the exhaustion of the forced CCW byte count (24 bytes), command chaining is initiated and the CCW that has been read into main storage location 08 is loaded into the channel. When the control unit presents 'channel end' at the completion of the CCW chain, a CSW store cycle is taken to load the I/O address into bits 21 to 31 of the word in location 00.

This ends the channel IPL operation. A 'load PSW' operation is then commenced during which the PSW is loaded from the double word in location 00. The CPU then commences operation as specified by the loaded PSW.

IPL Execution

System Reset

- System reset is intiated when the Load pushbutton is operated.

To commence an IPL operation on channel 0, the operator sets the LUA switches to an address that corresponds to a control unit on channel 0. The load pushbutton is then operated and a system reset is initiated; the system reset operation is described in Section 2.9.2 of "Console" in FEMM IBM System/360 Model 44, Form Y33-0007. At the completion of the system reset, the line 'channel start IPL' conditions the channel decode circuits in the common channel. If a channel 0 operation is decoded, the 'I/O execute' latch is set. 'I/O execute' and 'not data flow occupied' produce the line 'set address' which gates the address from the LUA switches into the channel address register for use during the initial-selection sequence.

Load Subchannel

- UCW 3 of the selected subchannel is loaded with the address of the next CCW (08).

- UCW's 1 and 2 are loaded with the forced IPL CCW.

'Set address' starts an interface clock cycle. At clock pulse S2, the 'UCW fetch request' latch and 'word 3 request' latches are set. An I/O cycle is taken, during which UCW 3 is read out to the D and E registers. The UCW's were previously reset during the system reset; therefore both D and E

registers contain zeros. The value of the D register is then increased by eight to produce the address of the next sequential CCW (08). A UCW 3 store cycle is taken to store the contents of the D register into UCW 3.

A CCW-fetch cycle is taken, gating the address contained in the E register (zero) to the SAR. Main storage location 00 is read out; this contains all zeros which are gated into the command and channel data registers. However, bit 6 of both registers is forced by 'generate IPL CCW', which sets an IPL read command into both registers.

A second CCW-fetch cycle is then taken. This time, the address contained in the E register is gated to the SAR, but bit 29 is forced so that the second word of the IPL CCW is read out (location 04).

During this cycle, CC and SLI flags are set. A byte count of 24 is gated into the E register.

Initial Selection

- The device is selected by an initial-selection sequence.
- The command is sent to the device.

The initial-selection sequence used during IPL is the same as that used during the start I/O operation. Refer, therefore, to "Start I/O Execution."

The address of the device is sent to the interface. When the addressed control unit replies with 'address in', the channel sends the IPL read command to the control unit. If the control unit sends back a status byte of zero, the operation continues; if not, the operation must be recommenced.

The channel then waits for the control unit to request service with a 'service in' tag.

Read 24 Data Bytes

- Twenty-four data bytes are transferred to main storage unless the control unit presents 'channel end' before the byte count is zero.

When the control unit presents 'service in' to the channel, the data transfer takes place in the same way as described previously in "Data Service."

The transfer of data bytes continues normally until either a byte count of zero is encountered or the control unit presents 'channel end' and 'device end'. No wrong-length-record indication is given because the SLI flag has previously been set.

Command Chain

- A command-chaining sequence commences when the initial CCW operation is completed.
- The CCW address in UCW 3 is fetched, loaded into the channel and updated.

When the initial CCW byte count (24) is exhausted, or the device signals 'channel end' and 'device end', a normal command-chaining operation is commenced.

'UCW 3 update' is generated which causes UCW 3 to be fetched, and the CCW address that it contains is set into the D and E registers. The value in the D register is increased by eight and the new CCW address is stored in UCW 3.

The CCW address (08) is then used during the two I/O cycles; which are taken to fetch UCW's 1 and 2. The operation is recommenced using the CCW that was read into location 08 during the transfer of the first 24 bytes of data.

IPL Termination

- IPL is terminated when a CCW with no chain flag is completed.
- A CSW store cycle is taken to store the channel, subchannel and device address into location 00.
- A load PSW operation is executed.

The sequence of data transfers and chaining continues until a CCW that does not contain a chain flag is encountered and executed. When 'channel end' is received for this CCW, a CSW store cycle is taken. The address 00 is generated and bits 21 to 31 of the SDR are loaded with the channel, subchannel and device address.

During this C-cycle, 'channel 0 and IPL' is generated, which starts a 'load PSW' operation. This loads the double word from location 00 as the new PSW. Details of the load PSW operation are contained in Principles of Operation - Processing Unit, Form Y33-0002, under "Status Switching Instruction."

The CPU then uses this PSW to execute the first instruction of the program.

I/O INTERRUPTS

- I/O interrupts ar requested by the channel which then waits for interrupt acceptance from the CPU.
- The CSW is stored during the interrupt routine.

The purpose of an I/O interrupt is to notify the program of the completion of an I/O operation. The interrupt is normally requested by the channel when it receives either 'channel end' or 'device end'. The setting of the interrupt request latch is detailed previously in "Start I/O Operation."

If the request is for a channel-end interrupt, UCW 1 is reset and the CSW is stored. If the request is for a device-end interrupt, the device is selected,

its status is stored in UCW 2 and the CSW is then stored.

The channel interrupt action occurs at the end of the new-PSW 1 period of the CPU interrupt procedure.

Channel 0 has the lowest priority of the channels for interrupts.

Interrupt Sequence

Interrupt Initiation

- During the new-PSW 1 period, the channel receives interrupt acceptance.
- Interrupt is cancelled if 'data flow occupied' latch is on for a device-end interrupt.

During the first interrupt cycle (new-PSW 1 period), the channel receives 'channel 0 interrupt accepted'. If the request is for a device-end interrupt, the 'data flow occupied' latch is tested. If this latch is on, the 'interrupt request cancel' latch is set. This action prevents the CPU from being held up by a channel operation. The 'end execute' latch is then set and the interrupt sequence is terminated.

The sequence that occurs after this time depends on the type of interrupt that has been requested.

Device-End Interrupt

- The device is selected by an initial-selection sequence.
- The status byte is set into UCW 2.

If, when the interrupt is accepted, a 'device end' is pending, the 'select interrupt' latch is set. This causes an initial-selection sequence to be started as described in "Start I/O Operation."

The address contained in the interrupt buffer register is set into the CAR and the initial-selection sequence continues. As the device has an interrupt condition pending, its response to 'select out' is 'status in', accompanied by a status byte of bus in. The status byte is loaded into the channel data register and an I/O cycle is requested.

The zero control latch is set and UCW 2 is read into the SDR. The device status is loaded from the channel into the SDR and stored in UCW 2. The CSW is then stored as described later in this section.

Channel-End Interrupt or PCI

- UCW 1 is reset before the CSW is stored for a channel-end interrupt.
- The CSW store cycles are entered directly for a PCI.

If, when the interrupt is accepted, a channel-end interrupt is pending, a UCW 1 reset cycle is taken. The interrupt buffer register address is gated to the SAR and UCW 1 is read out to the SDR. UCW 1 is then loaded with zeros and stored and the CSW cycles are commenced.

For a PCI, UCW 1 is not reset and the CSW store cycles are started immediately.

CSW Storing

- Two C-double cycles are taken.
- UCW 3 is loaded into CSW 1.
- UCW 2 is loaded into CSW 2.

CSW's 1 and 2 are located in main storage locations 40 and 44 hex respectively. In order to store the CSW's, two C-double cycles are taken. A C-double cycle cannot be interrupted between its two storage cycles since the SDR is used as a temporary storage unit between cycles.

During the first C-cycle of the CSW 1 double cycle, UCW 3 is read out of storage (its address being gated from the interrupt buffer register) and it is set into the SDR. A second cycle is then taken but the reset of the SDR is blocked. During this cycle, the address of CSW 1 (40 hex) is loaded into the SAR but the set of SDR is blocked. During the write section of the cycle, the data that originated from UCW 3 is set into location 40 hex.

The second C-double cycle is similar to the first except that UCW 2 is read out and its content loaded into CSW 2 (44 hex).

Load Interrupt Code

- The device address from the interrupt buffer register is loaded into the old-PSW interrupt code.

During the old-PSW 1 period, the line 'store interrupt code' is generated. This gates the interrupt buffer register contents into bits 24 to 31 of the old PSW 1. At WC1 of this cycle, the interrupt buffer register is reset and the channel-interrupt sequence is complete.

CHECKING

Parity Checking

- Parity is not carried by channel 0.
- Parity is checked on bus in.

The data and address paths through channel 0 do not carry parity bits. However, whenever data is set to bus out, correct parity is generated (odd) and data received on bus in is parity checked.

Channel 0 Check Latches

- Five check latches are provided in channel 0.
- Two latches record machine checks.
- Three latches record program or data errors.

The five check latches (Figure 3B-34) provided in channel 0 to record machine, program or data errors are:

Control check latch
Interface control check latch
Program check latch
Incorrect length latch
Data check latch.

Control Check Latch

The control check latch records errors that occur and which indicate a channel malfunction. Three setting inputs are provided to the latch; two of these are error conditions, the third is provided to set the latch if a channel-control check has been recorded in UCW 2 during a previous operation. The two error conditions that cause this latch to be set are:

An SDR parity check on UCW or CCW cycles.
A cycle-control check on channel 0 cycles.

The output of this latch is 'channel status control check' which requests an external machine-check interrupt.

Interface Control Check Latch

This latch records invalid conditions caused by errors on the standard interface. It is set by any of the following conditions:

1. Any two interface in tags active at the same time.
2. An invalid device address presented by a control unit.
3. The address on bus in not matching the address on bus out during initial selection.
4. 'Bus in bad parity' active during 'address in' or 'status in'.
5. UCW 2 bit 14 (interface control check) set during a previous operation.

The output of this latch requests an external machine-check interrupt.

Program Check Latch

The program check latch is set by any invalid conditions that occur as a result of program errors. The conditions that set the latch can be seen in Figure 3B-34. Typical conditions include invalid CAW or CCW formats.

Incorrect Length Latch

The incorrect length latch is set if any I/O operation is terminated before the byte count has been exhausted. The setting is conditioned by the SLI flag.

Data Check Latch

The data check latch is set by either bad parity on bus in during a 'service in' sequence, or a storage data check during an I/O operation.

CLEAR EXTENSION STORAGE

- Extension storage is cleared as the result of a channel 0 check.
- The interrupt buffer register is loaded with the address of the first UCW (all ones).
- I/O cycles are taken to reset all UCW's, the value in the interrupt buffer register being reduced by one on each cycle.
- The operation ends when the interrupt buffer register contains all zeros.

When, during an interrupt cycle, the CPU sends 'clock accepted for channel 0' to the channel, the 'clear extension'latch is set and the interrupt buffer register is set to all ones. An I/O cycle is requested.

During this I/O cycle, the first UCW is cleared to all zeros. The value in the interrupt buffer register is then reduced by one and another I/O cycle is requested, during which the same action takes place. This routine is continued until all UCW's have been reset, at which time the interrupt buffer register contains all zeros and the 'clear extension' latch is reset.

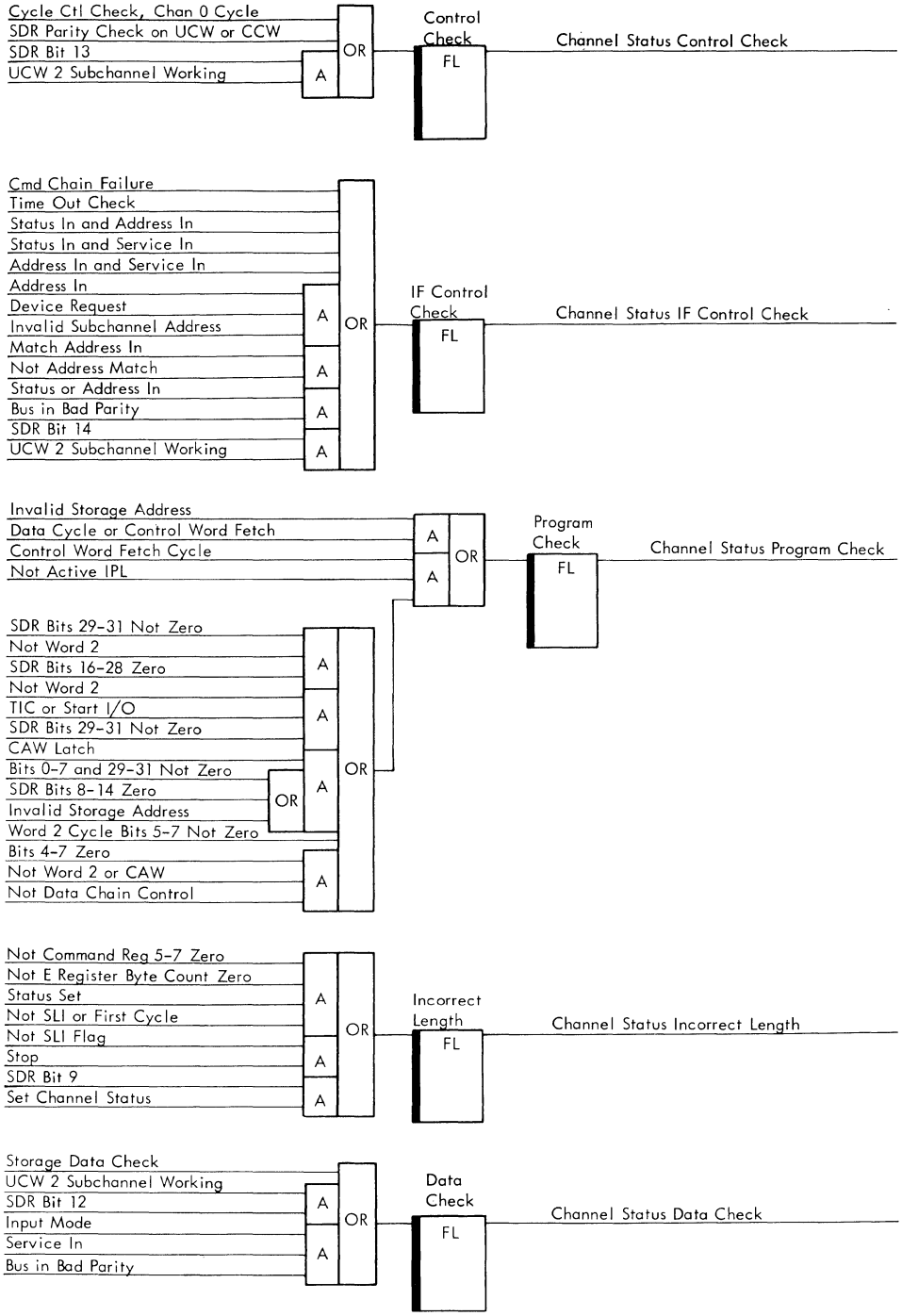


Figure 3B-34. Channel 0 Check Latches

HIGH SPEED MULTIPLEXOR CHANNEL

DESCRIPTION

- Channel controls the transfer of information between I/O devices and the CPU.
- Channel is controlled by the CCW.

A channel is used to feed the CPU with information from the input devices. The channel also feeds information from the CPU to the output devices.

All information necessary for the channel to perform an input or output operation is fetched from the CCW. As soon as this information is loaded into the channel by the CPU, the channel controls the entire I/O operation, and the CPU continues with the next sequential instruction.

Performance

- Two HSMPX channels may be connected to System/360 Model 44.
- Each channel may have up to four subchannels.
- The HSMPX channel can work in either burst mode or byte mode.
- Two bytes may be transferred between main storage and the channel at the same time.
- One byte at a time is transferred between the channel and the I/O device.

Figure 3B-35 shows a simplified data flow of one HSMPX channel. The 32 + 4 bits data bus out from the SDR is decreased by the halfword select to a 16 + 2 bits data bus in the channel. This two-byte bus is connected to all the subchannels. B0, B1 and B2 registers are used as data buffers between the CPU and the devices.

The UCW is loaded during the execution of the start I/O instruction, and contains all the necessary controls to perform the I/O operation. The interface control is common to all subchannels.

The HSMPX channel is capable of operating up to four medium-speed I/O devices simultaneously with a minimum of main storage or program interference.

The channel can work either in burst mode or byte mode. In burst mode, the selected subchannel and device remain connected until the entire I/O operation is completed; during this time, no other subchannel or I/O device can be operated. In byte mode, however, all four subchannels and their selected devices are able to work simultaneously by using the interface one at a time.

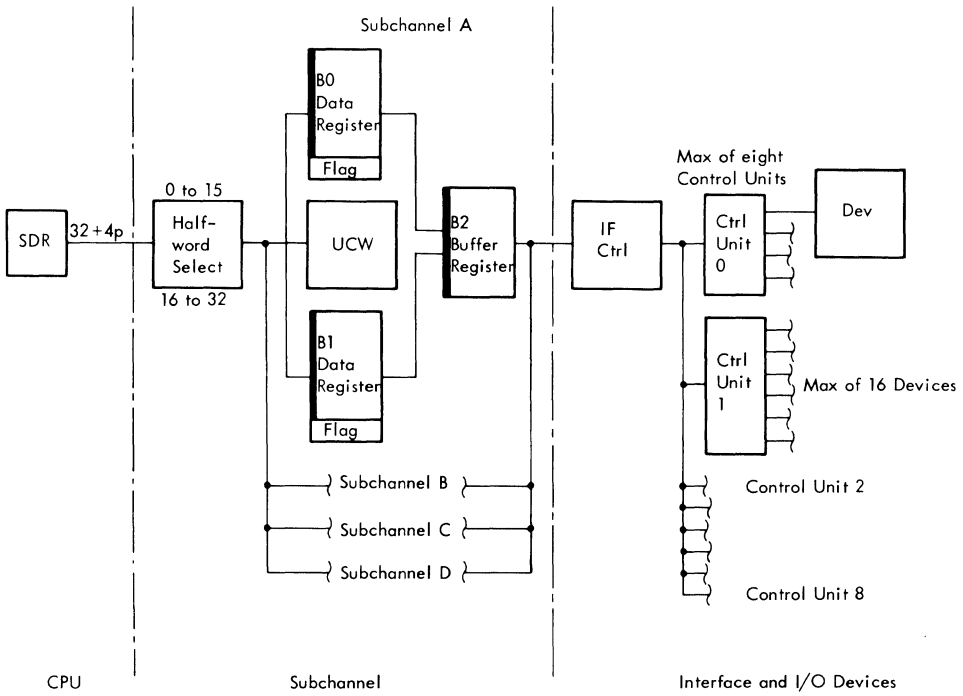


Figure 3B-35. HSMPX Simplified Channel Data Flow

Priority

- HSMPX channel has a higher priority than MPX channel 0.
- Priority between the two high-speed channels is shared equally.
- Priority between the subchannels is on an A, B, C, D basis.

If the Model 44 is fitted with both an MPX channel 0 and an HSMPX channel, the HSMPX channel has the high priority.

If, however, two HSMPX channels are connected to the same system, both channels have the same priority. For example, if channel 1 has the last I/O request and both channels send the next I/O request at the same time, then channel 2 has the higher priority.

Subchannel A has the highest priority and subchannel D the lowest priority among the subchannels.

A chaining operation and an I/O request for data transfer has priority over start I/O operations.

Capacity

- Each subchannel is able to store a maximum of three bytes of data.
- Each subchannel may be connected to a maximum of two control units.
- Each control unit may be connected to a maximum of 16 devices.

A maximum of three data bytes may be stored in each subchannel, two bytes in B0 and B1 data registers and one byte in B2 buffer register.

A maximum of two control units may be connected to each subchannel; this means that the maximum number of control units on one channel is eight. Each control unit can be connected to a maximum of 16 devices but the actual number of devices is limited by the control unit used.

DATA FLOW

The HSMPX channel data flow is shown in FEMD, Figure 1003.

Data Address Register

- Eighteen bits wide.
- Designates the data in main storage to be used during the I/O operations.
- Value in the register is increased during read forward and write, and reduced during read backward.

The Data Address Register (DAR) is an eighteen-bit register used to designate the location in main storage from which data is taken, or into which data is loaded during an I/O operation.

Bit 14, which is the leftmost bit of the register, is used only for checking purposes. When this bit is on, and the address is set in the SAR in the CPU, an invalid address results.

Bits 30 and 31 of the DAR are used to control the data flow within the subchannel. Bit 31 is used to gate the B2 buffer register into either the B0 or B1 data register. Bit 30 is used to control the 16-bit data path between the B0 and B1 registers and the CPU SDR.

The value in the DAR can be increased or reduced by two: this is done by feeding a pulse to bit 30 and propagating any carry to bits 29, 28, 27 and so on. Incrementing or decrementing takes place every time data is stored or fetched during a storage cycle.

Bit 31 is also changed each time a data byte enters the B2 buffer register from bus in or leaves B2 for bus out. The changing of bit 31 has no effect on the reset of the DAR.

The DAR is loaded during fetching of the CCW.

Command Address Register

- Fifteen bits wide.
- Bits numbered from 14 (leftmost) to 28.
- Designates the location in main storage from which the CCW is to be fetched.
- Value in the command address register can be increased by eight.

The 15-bit Command Address Register (COAR) designates the location of the CCW in main storage. Since a CCW is always located on a double-word boundary in main storage, bits 29 to 31 are unnecessary. Bits 14 to 28 are loaded from the SDR during a start I/O operation.

When CCW 1 is fetched, bits 29 to 31 in the SAR are zero. However, in order to fetch CCW 2, a one is forced in bit position 29 in the SAR from the I/O or chaining counter. After the fetching of CCW 2, the value in COAR is increased by eight.

Count Register

- Sixteen bits wide.
- Designates the number of bytes transferred to or from storage for the I/O operation.
- The count must never be zero at the initiation of the start I/O operation.
- The count register is loaded from the CCW in main storage.
- The count is reduced by one during read or write operations.
- The maximum number of bytes to be transferred for one start I/O, without data chaining is 65,536.

Op Register

- Op register is set to determine which operation is to be performed.
- Op register can be set to:
  Read forward
  Read backward
  Write.

The Op register is a two-bit register used to indicate which type of operation is to be performed for the current operation. Effectively, for the subchannel, there are only the three types of operations listed previously.

Figure 3B-36 shows which operation is set for the different commands.

	Command Bits				Op Reg Bits		Subchannel
Command	4	5	6	7	0	1	Operation
Sense	0	1	0	0	1	1	Read
TIC	1	0	0	0	-	-	-
Read Backward	1	1	0	0	1	0	Read Backward
Write	M	M	0	1	0	1	Write
Read	M	M	1	0	1	1	Read
Control	M	M	1	1	0	1	Write

M = Modifier Bit

Figure 3B-36. Op Register Settings for Commands

B0 and B1 Data Registers

- Eight bits wide plus one parity bit.
- Used mainly as buffers for data transfer.
- Each register has an additional flag bit which indicates that the register has a data byte stored for a read operation.
- All data from main storage is inverted when loaded into the data register.

Both registers have a capacity of eight bits plus one parity bit. The registers are used mainly as buffers for data to be transferred between main storage and the devices; two bytes of data can be transferred between the data registers and the main storage at the same time. Additionally, each register has a flag bit that is on when the register is full (register contains a byte of data) for a read operation.

The B0 data register positions are numbered from 0 to 7 and those of the B1 data register from 8 to 15.

B2 Buffer Register

- Eight bits wide plus one parity bit.
- Used mainly as a buffer for storing data.

The B2 buffer register is similar to the data registers except that it has no flag bit. This register is used as a buffer for data between the interface bus and the two data registers.

Flag Bits

- Flag bits are loaded from the CCW in main storage as follows:
  CD flag : Chain data flag
  CC flag : Chain command flag
  SLI flag : Suppress-length-indication flag
  SKIP flag : Skip-transfer-of-data flag
  PCI flag : Program-controlled-interruption flag.
- The 'end' flag is set when channel-end status is received from a control unit.

For full details of the flag bits, see IBM System/360 Principles of Operation, Form A22-6821.

Subchannel Status

- The following conditions are detected and indicated by the subchannel:
  - Wrong length record
  - Program check
  - Channel data check
  - Channel control check
  - Interface control check.

For details, refer to the subsequent "Checking" section in this manual.

Device Address Register

- Five bits wide.
- Holds control-unit identifier and device address.

This is a five-bit register, with the bits numbered from 3 to 7. Bit 3 identifies one of the two control units within the same subchannel. Bits 4 to 7 identify the device address.

The Device Address Register (DEVAR) is loaded from the control unit via the interface. During an I/O interrupt, the DEVAR, the subchannel address and the channel address are set into the PSW as part of the interrupt code.

Compress

The standard address register for System/360 is 24 bits wide, numbered from 8 to 31. On the System/360 Model 44 however, the largest storage capacity is 128K bytes; therefore, bits 8 to 14 must always be zero. Bits 8 to 14 are gated into an OR circuit known as the compress. The output is then gated to bit 14 in the COAR and the DAR.

Byte Control

The byte-control logic is used to determine which bytes of the SDR are loaded and which bytes are regenerated during a read operation and for the storing of the CSW.

For a read operation, the following conditions effect the byte control:

B0 and B1 data register flag bits
Bit 30 of the DAR.

When a CSW is stored, the I/O counter and the current I/O operation control the byte control. Figure 3B-37 shows the effect of a typical byte control on the SDR. Byte control 0 controls byte 0 in the SDR; byte control 1, byte 1 in the SDR; byte control 2, byte 2 in the SDR and byte control 3, byte 3 in the SDR.

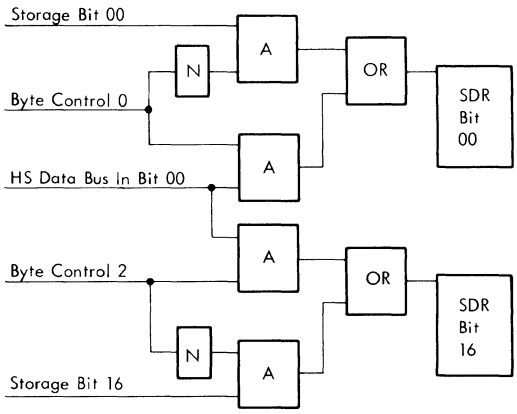


Figure 3B-37. Byte Control Gating

When byte control 0 is active, bits 0 to 7 of the data bus in are gated to bit positions 00 to 07 of the SDR. However, if byte control 0 is not active, the contents of the storage are gated into bit positions 0 to 7 of the SDR and this data is regenerated. The reset of the SDR is normal (all bytes in the SDR are reset before new data is gated).

Since the data bus in is only a 16-bit path, bit 00 on the data bus in is connected to SDR bits 00 and 16, bit 01 on the data bus in to SDR bits 01 and 17, and so on.

SUBCHANNEL COMMON CONTROLS

During loading of information from the CAW and the CCW into the subchannel, different controls are necessary. Since only one start I/O can be executed at one time, these controls can be used for all four subchannels and for both channels.

A chaining operation, however, can be initiated on one subchannel at the same time as another subchannel is executing a start I/O. To allow for this, some multiple controls are used only for chaining.

Channel Address Word Latch

The CAW latch is set to indicate that the CAW is fetched from main storage.

CAW Cycle Latch

This latch is set by the CAW latch and the 'C-cycle control' latch.

I/O Counter

The logic and timing of the I/O counter are shown in Figure 3B-38. This two-bit counter is used as a sequence control during fetching of the CAW and CCW, and during storing of the CSW.

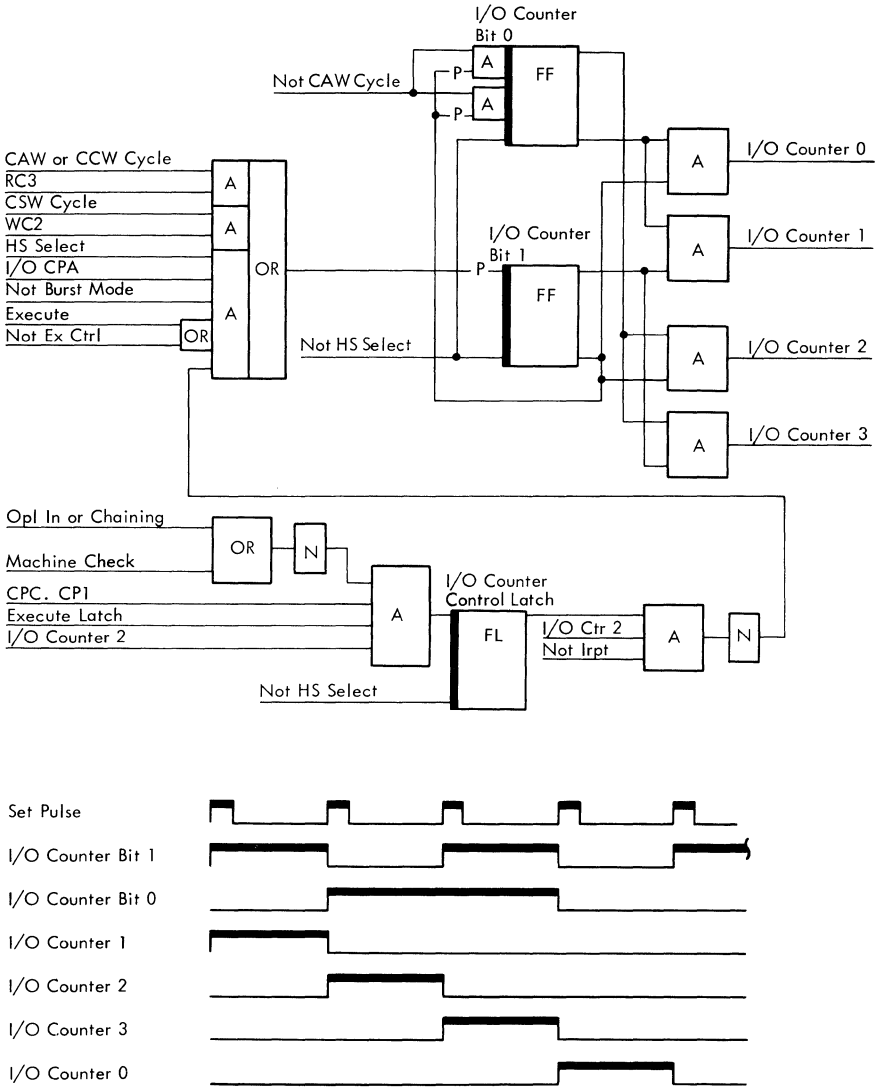


Figure 3B-38. I/O Counter

The I/O counter also provides timing pulses to analyze the condition of the selected subchannel for any I/O operation. If 'operational in' is up, a chaining routine is in process. If machine-check reset is performed at the time any I/O operations are started, the I/O counter holds at I/O counter 2 until the operation may again proceed. During an I/O interrupt, the I/O counter is also advanced in order to analyze the selected subchannel and the state of the interface. During the CAW-fetch, the I/O counter bit 0 is inhibited from being turned on.

CCW Latch

This latch is set during the last CAW cycle and stays on during the fetching of the CCW for a start I/O execution.

CCW Cycle Latch

This latch is set by the CCW latch and the I/O cycle control latch. The output of this latch is AND'ed with the I/O counter and gives I/O counter cycles 1 to 4.

Chaining Counter

This counter is similar to the I/O counter. If a Transfer-In-Channel (TIC) command is detected, the counter is stepped only from 0 to 1.

CCW Cycle Latch for Chaining

This latch is on when the CCW is fetched during a chaining operation. The output of the latch is

AND'ed to the chaining counter and gives chaining-counter cycles 1 to 4.

Interface Delay-Line Drive

The interface delay-line drive logic and timing are shown in Figure 3B-39.

The interface-tag driver is used for timing the data passing the interface, and for gating data between the B0 and B1 data registers and the B2 buffer register.

The interface delay-line drive consists of three 125-nanosecond time-delay drivers. The delay-line drive is started when one of the following conditions is present:

'Select in' tag up.
'Service in' tag up and B2 buffer register is empty for read and full for write.
Gated 'address in'.
'Status in' tag up and B2 buffer register is free.

The outputs of the three time-delay drivers are AND'ed together to give different timing pulses (Figure 3B-39).

CPU Subchannel Select and Interrupt Priority

The logic for the CPU subchannel select is shown in Figure 3B-40. This circuit is used to select the subchannel during the following:

Start I/O
Test I/O
Halt I/O
An initial program loading
An interrupt.

For start I/O, test I/O and halt I/O, bits 25 and 26 in the B register are used to select the subchannel.

During an IPL operation, the address bits 25 and 26 from the LUA switches on the console select the subchannel. At the time an interrupt is accepted, however, the subchannel is selected according to the state of the 'subchannel interrupt request' latch. If more than one 'subchannel interrupt request' latch is on at the same time, the subchannel with the highest priority is selected first. This priority is controlled by the interrupt priority circuit shown in Figure 3B-40.

Interface (IF) Subchannel Select Register

The IF subchannel select register is shown in Figure 3B-41. This is a two-bit register used to select one of the subchannels to the interface. The two bits can be set from one of the two following sources:

IF bus in bits 01 and 02.

The output of the IF request priority circuit. During an initial-selection sequence (channel select control unit) the IF subchannel select register is always set from the channel. During a request-in sequence (control unit select subchannel), however, the bits are set from bus in.

IF Subchannel Select Decode

The IF subchannel select decode is shown in Figure 3B-41. The output of the two bits of the IF subchannel select register is decoded to select one of the four subchannels.

IF Request Priority

This circuit is shown in Figure 3B-41. The circuit selects subchannels according to either the state of any of the chaining 'IF request' latches or the output of the interrupt priority circuit (Figure 3B-40).

Units Address Match

This five-bit compare circuit is used for comparing bits 27 to 31 in the B register with the bits in the DEVAR during a 'test I/O' operation. The circuit is common to both channels.

IF Address Match

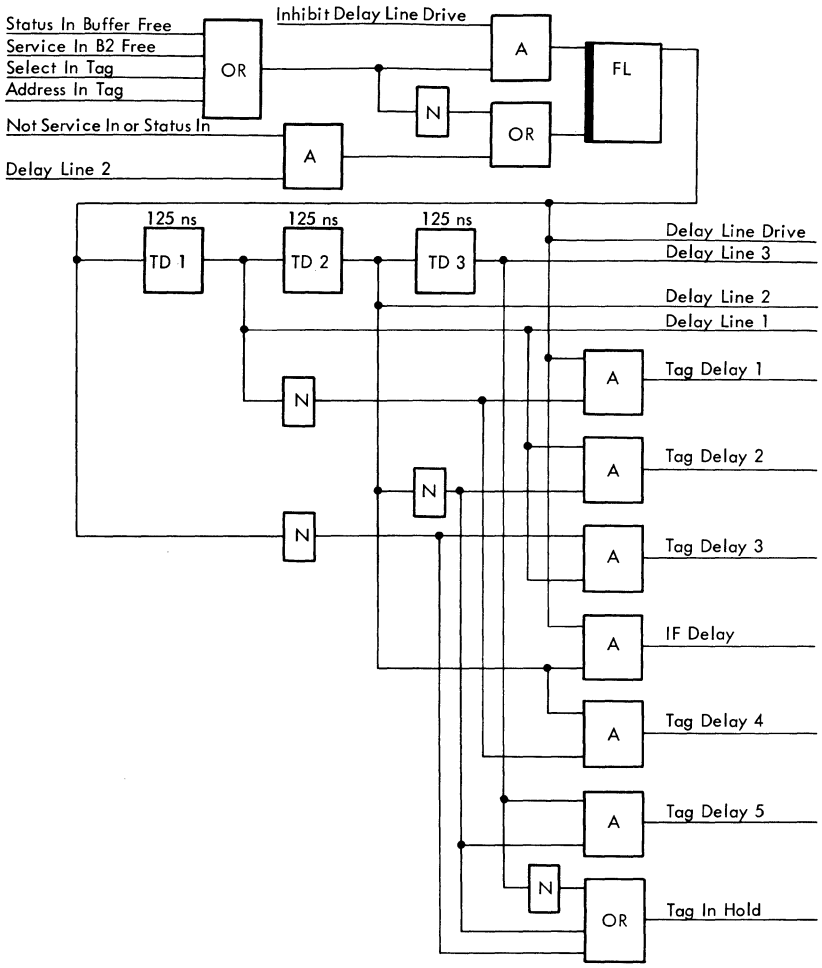
This eight-bit compare circuit is used for comparing the address on bus out with the address on bus in during initial selection. The same circuit is also used to test for a status-equal-to-zero condition (all bits on bus in are zero). The logic of the circuit is shown in Figure 3B-42.

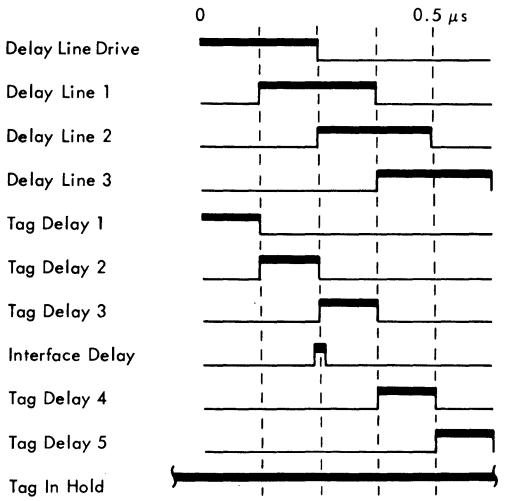
Bit 3 Match

This one-bit compare circuit is used for comparing bit 3 in the DEVAR with bit 3 on bus in during a request-in-for-status sequence. The function ensures that the control unit that sent 'request in' is the control unit currently operating with the selected subchannel. The logic associated with this function is shown in Figure 3B-42.

Bus-Out Parity Check and Generate

Bits 0 to 7 on bus out are gated to an ODD circuit. The output of this circuit is gated together with the parity bit to an Exclusive OR (EXOR) circuit to perform the parity check. Correct parity is always generated.





Timing Chart is shown for Address In or Select In Conditions.

Figure 3B-39. Delay-Line Drive and Timing

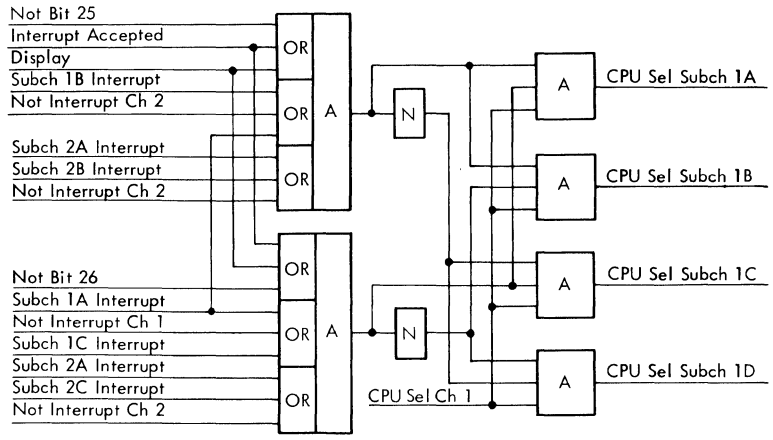


Figure 3B-40. CPU Subchannel Select and Interrupt Priority

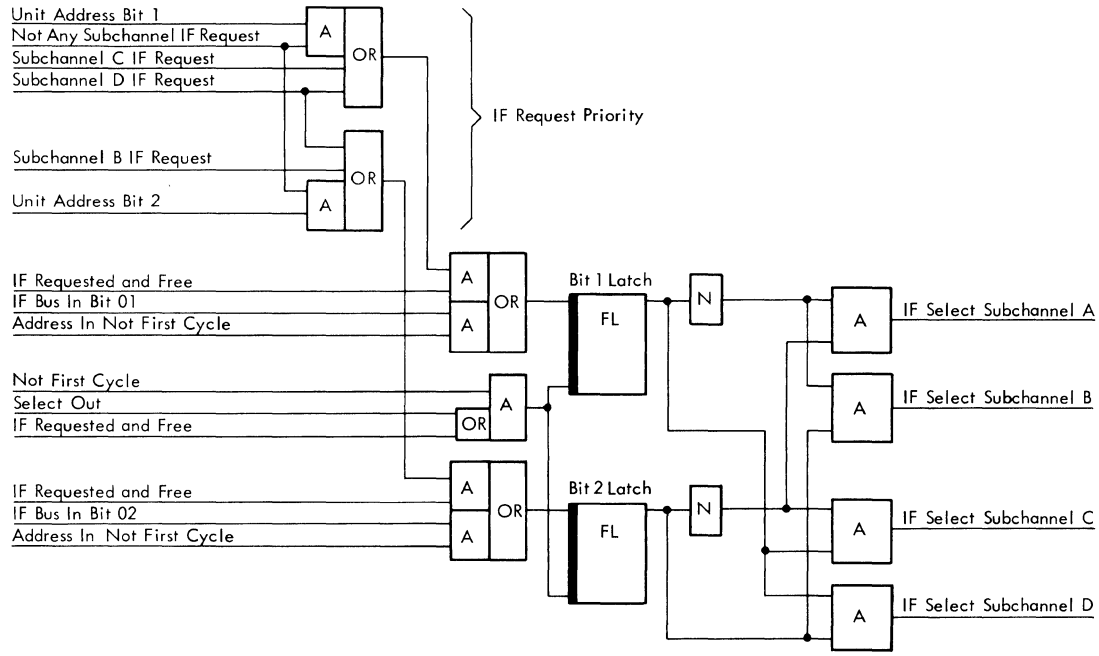


Figure 3B-41. IF Subchannel Select Register, Select Decode and Request Priority

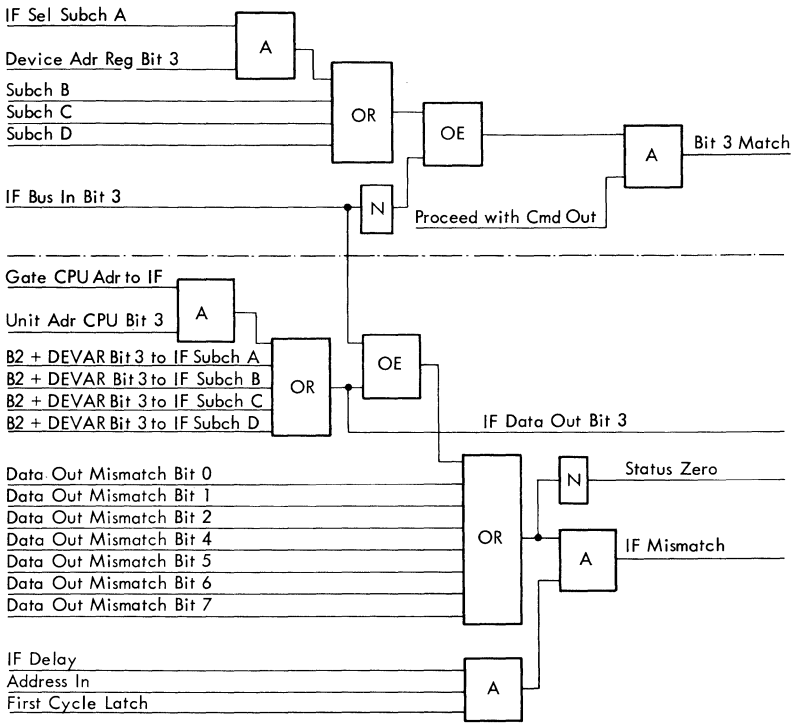


Figure 3B-42. IF Address Match, Status Zero and Bit 3 Match

Bus-In Parity Check

This circuit is similar to the bus-out parity check circuit.

Halfword Select

The halfword-select and control logic is shown in Figure 3B-43.

Since the data bus to and from the subchannel is 16 + 2 bits wide, only one halfword can be transferred to and from the SDR at one time. The halfword select, which controls the data bus out, can therefore gate either bytes 0 and 1, or bytes 2 and 3, to the subchannel. The halfword select control is common to both channels; there is, however, one halfword select for each channel. The halfword select also gates the bits necessary during an IPL operation.

The halfword select used on data bus in works with byte control (see "Byte Control").

Burst Mode Latch

Each channel has a burst mode latch which indicates that the interface is working in burst mode.

The burst mode latch is set, if 'operational in' stays up for more than 100 microseconds, by a delay circuit and three flip-flop triggers (Figure 3B-44). The delay circuit consists of two 10-microsecond singleshots, connected as a free-running oscillator which continues to operate except during system reset.

The output of singleshot 1 is gated to flip-flop 1. The output of flip-flop 1 is gated to flip-flop 2, and so on; therefore, flip-flop 2 is set when flip-flop 1 goes off, and flip-flop 3 is set when flip-flop 2 goes off. The flip-flops, however, are not set until the timer control latch is set on. This latch is set by either 'address out' or 'address in' and stays on until either 'operational in' falls or 'device end' is analyzed in the subchannel.

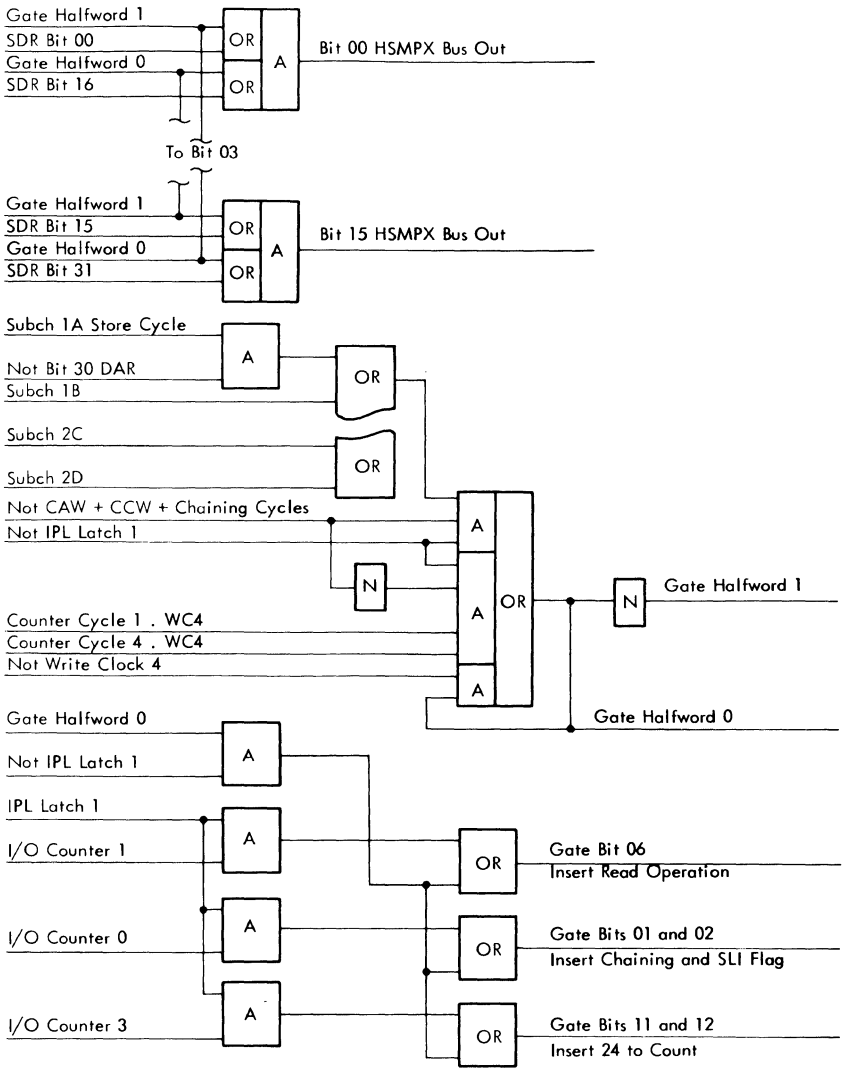


Figure 3B-43. Halfword Select and Control

For an initial-selection sequence (selection of a control unit initiated from the subchannel), the flip-flops start advancing when 'address out' is raised. At the end of the initial selection, the flip-flops are reset if 'operational in' drops. If 'operational in' stays up for more than 100 microseconds, the burst mode latch is set.

For a request-in sequence (a selection of the control unit initiated from that control unit), the flip-flops start advancing when 'address in' is raised.

Tag Time Out

'Tag time out' (Figure 3B-44) is a signal which indicates that a hang-up condition has occurred during an initial-selection or request-in sequence. 'Tag time out' is generated if 'status in' or 'service in' has not dropped 40 microseconds after 'address out' or 'address in' is raised. To provide for this delay, the singleshots and flip-flops used for the burst-mode latch control also control 'tag time out'.

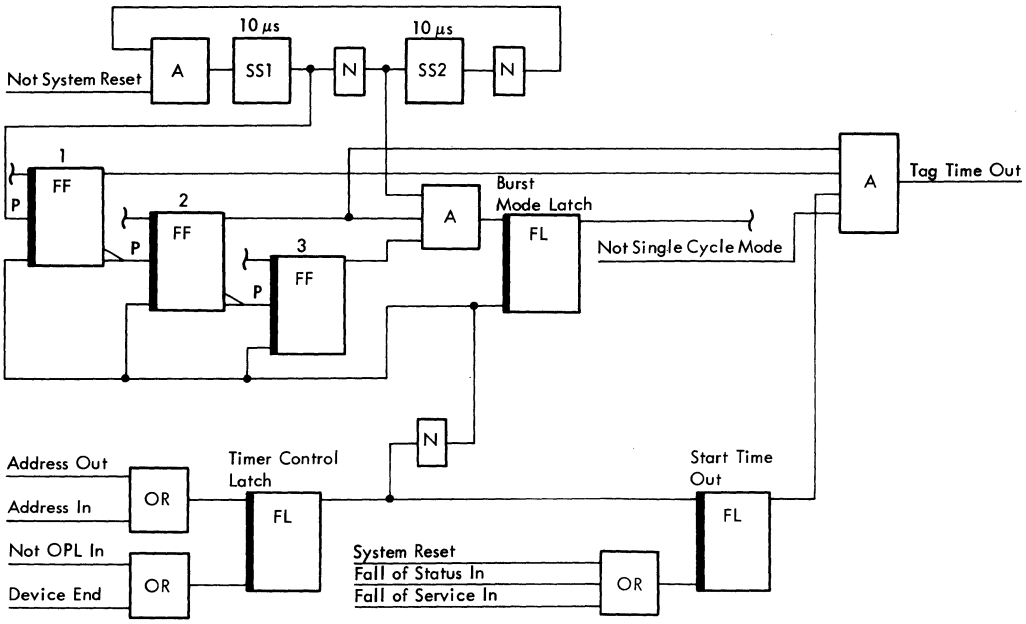
The timer control latch turns on the 'start time out' trigger; if this trigger is not reset before flip-flops 1 and 2 are on, 'tag time out' is gated and the 'IF control check' latch is set.

The start time out trigger is normally reset by the fall of either 'status in' or 'service in'.

Interface Single Cycling

For single-cycling through an operation on the interface, the gating of the following outbound tags is under the control of 'start key singleshot':

Address out
Command out
Service out.



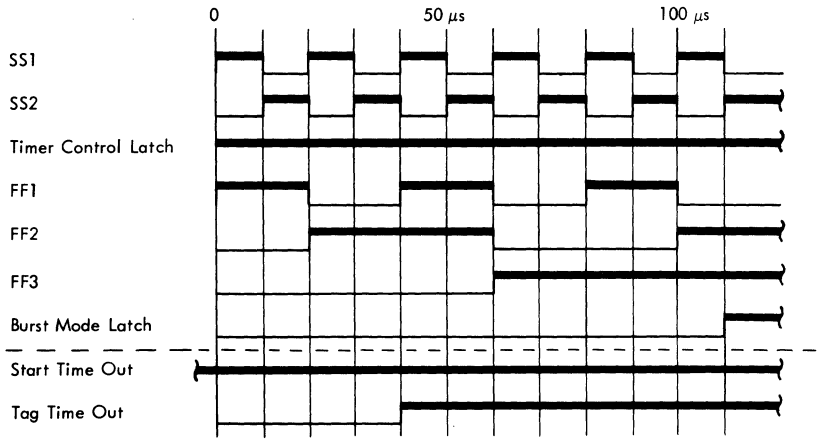


Figure 3B-44. Burst Mode and Tag Time Out

The interface single-cycling circuit is shown in Figure 3B-45. The single-cycle latch is under the control of the 'start key singleshot' if the console rate switch is set to SINGLE CYCLE. When the start pushbutton is depressed, the first singleshot turns on the single-cycle latch, the second singleshot turns it off and the sequence is repeated each time the button is operated.

Each of the three outbound tags has a 'single-cycle control' latch, which can be set only if the single-cycle latch is off and if the appropriate outbound latch is on. The 'single-cycle control' latch is on until the start pushbutton is again depressed.

Example of Single Cycling during an Initial-Selection Sequence

For this example, the following outbound tags are gated in the following sequences:

Address out
Command out
Service out.

Note that no inbound tag is gated before the outbound tag is dropped.

Assume that the console rate switch is set to SINGLE CYCLE and that the 'address out' latch goes on after the first depression of the start pushbutton.

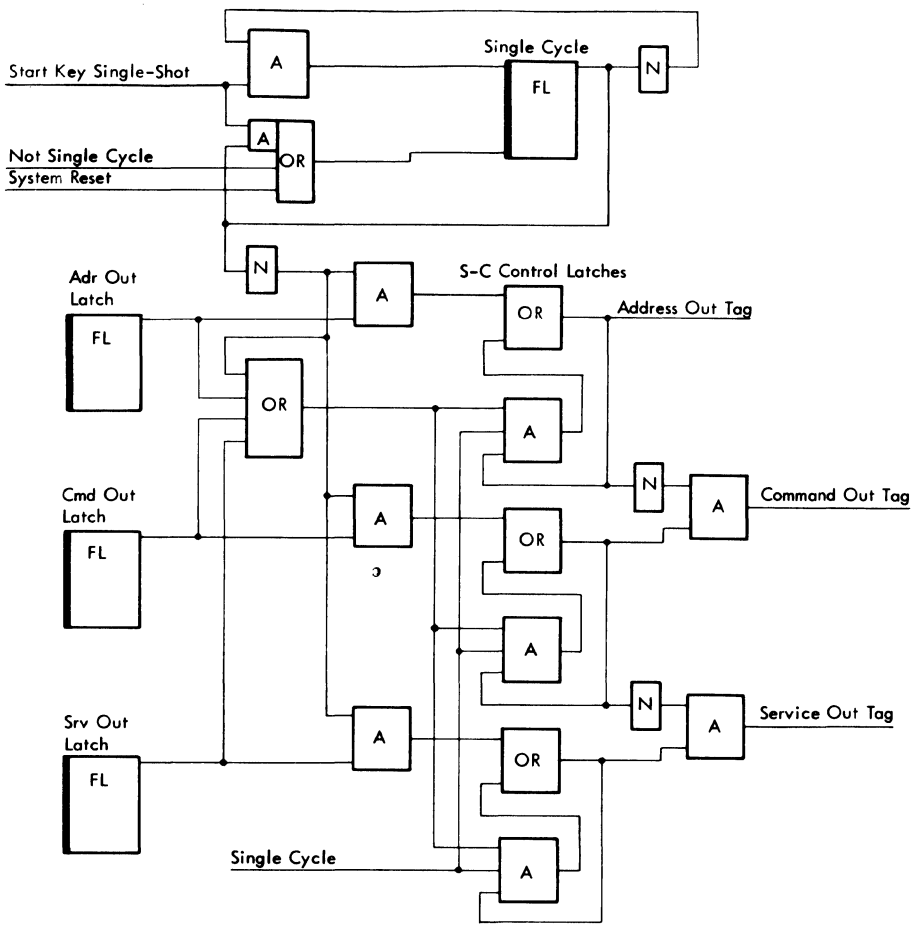


Figure 3B-45. Single-Cycle Latch and Tag Out Gating

The single-cycle latch is on at this time and prevents 'address out' from being gated to the interface.

When the start pushbutton is operated again, the single-cycle latch goes off and sets the address out 'single-cycle control' latch. The 'address out' tag is now gated.

The next singleshot from the start pushbutton turns on the single-cycle latch which again resets the address out 'single-cycle control' latch. The 'address out' tag is dropped. The dropping of the 'address out' tag causes the 'address in' tag to be gated which, in turn, set the 'command out' latch.

During the next depression of the start pushbutton, the 'command out' tag is gated.

The fall of the 'command out' tag during the following singleshot raises the 'status in' tag which sets the 'service out' latch.

On the next depression of the start pushbutton, the 'service out' tag is gated and remains up until the button is again operated.

The single-cycle timing for the service-in sequence is shown in Figures 3B-46 and 3B-47 for initial selection and a burst-mode operation respectively. The 'service out' tag is gated every other time that the start pushbutton is depressed.

START I/O OPERATION

The purpose of the start I/O instruction is to transfer data between any I/O device and the CPU. The type of data transferred may be either data information, sense information or control information, depending on the current command.

A start I/O operation can be divided into distinct steps as follows:

I-fetch
Start I/O execution
Data service
End procedure
Command chaining
Data chaining
Transfer-in-channel.

Figure 3B-46. Single-Cycle Timing for Initial Selection (Service-In)

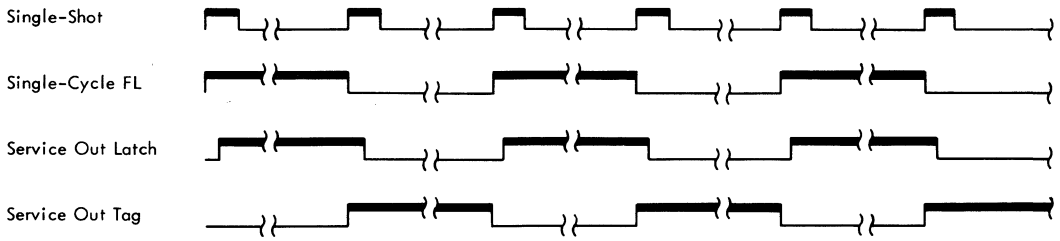


Figure 3B-47. Single-Cycle for a Burst-Mode Operation (Service-In)

I-Fetch

- Start I/O instruction is fetched.
- The channel, subchannel, control unit and device address is generated in the B register.

The start I/O instruction uses the SI format. The CPU must be in the supervisor state when the instruction is executed.

The I-fetch phase is described in detail in Principles of Operation - Processing Unit, Form Y33-0002. At the end of the I-fetch, the channel, subchannel, control unit and device address is available in the B register bits 21 to 31. The decoding of this address is shown in Figure 3B-48.

Bit 21 is not used and must always be zero. Bits 22 and 23 are used to select one of the three channels. Bit 24 must always be one for the HSMPX channel to indicate that the subchannel is shared with two control units.

Bits 25 and 26 are used to select one of the four subchannels and also the two control units belonging to the selected subchannel. Bit 27 selects one of the two control units within the same subchannel. Bits 28 to 31 select the I/O device.

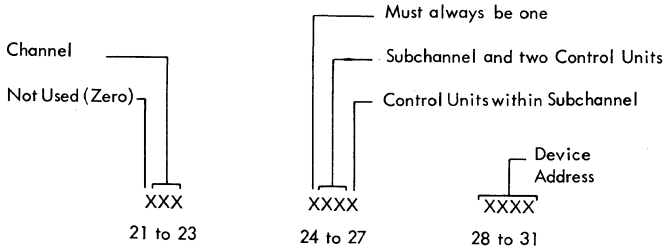


Figure 3B-48. Start I/O Address

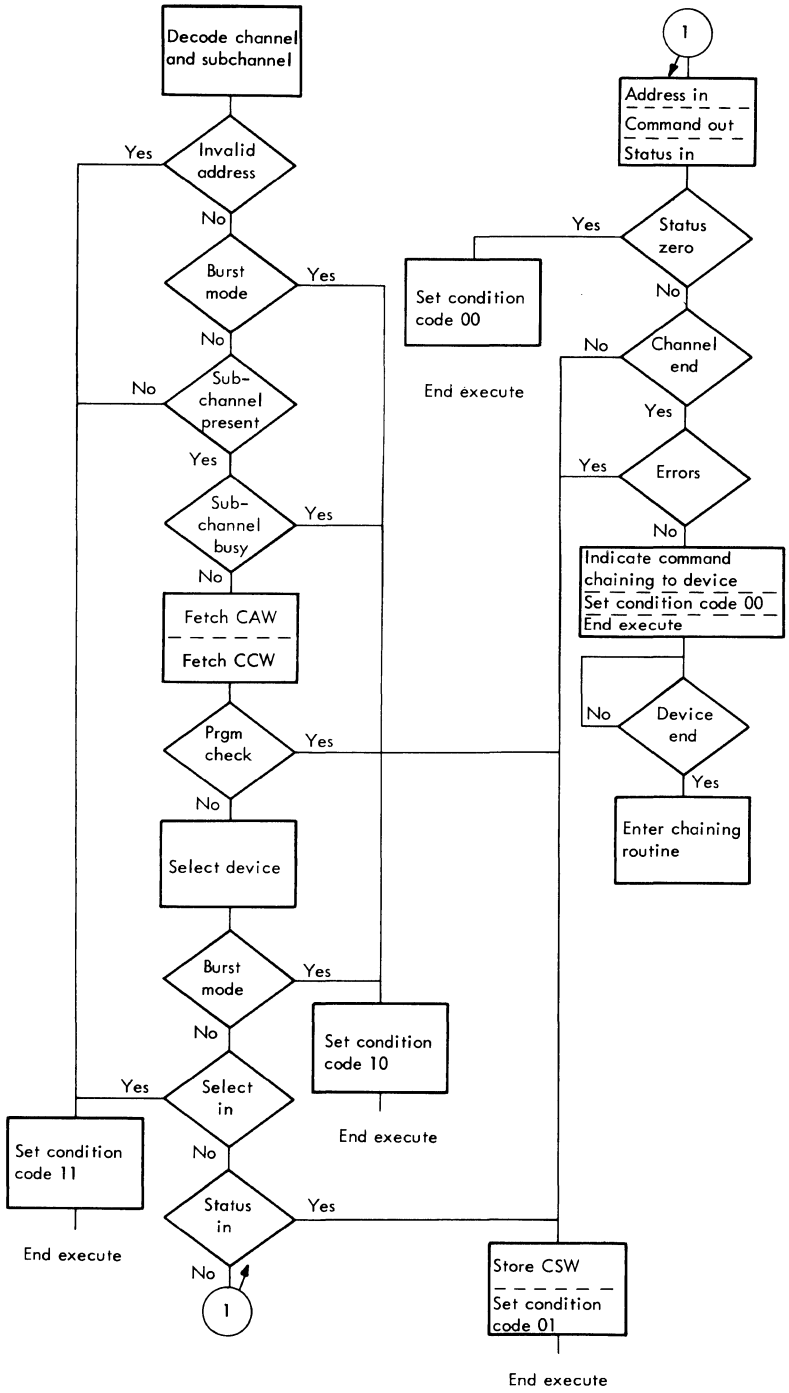


Figure 3B-49. Start I/O

Start I/O Execution

A simplified flow chart of start I/O execution is shown in Figure 3B-49. The start I/O execution period is the time during which the addressed channel, subchannel and device are set up to perform a data transfer. The start I/O execution is always ended by setting a condition code as follows:

00 : The start I/O operation is successfully started.
01 : The status portion of the CSW is stored.
10 : The subchannel or interface is busy.
11 : The channel, subchannel or device is not operational.

The start I/O execution can be divided into the following phases:

Channel and subchannel interrogation
CAW-fetch
CCW-fetch
Initial selection.

Channel and Subchannel Interrogation

- The compute clock is started to decode the channel and subchannel.
- Bits 22 and 23 in the B register select the channel.
- Bits 25 and 26 in the B register select the subchannel.
- The I/O counter is advanced by I/O clock pulses.
- The 'execute' latch is turned on at I/O counter 2 time.

At the end of I-fetch, the CPU sequence control latch starts the compute clock. At CC6, the channel decode latches are set according to bits 22 and 23 as shown in the following table.

Bits 22	Bits 23	Channel Selected
0	0	Channel 0
0	1	Channel 1
1	0	Channel 2
1	1	Invalid

If at this time an invalid address is detected, a condition code of 11 is set in the PSW and the operation is ended.

A successful selection of one of the HSMPX channels results in decoding of the subchannel and the I/O counter is started by I/O clock pulses. The subchannels are decoded by bits 25 and 26 as shown in the following table.

Bits 25	Bits 26	Subchannel Selected
0	0	Subchannel A
0	1	Subchannel B
1	0	Subchannel C
1	1	Subchannel D

The subchannel is now selected by the CPU until start I/O execution is completed (refer to Figure 3B-40).

At I/O counter 2 time, the execute latch is set in order to test the condition of the subchannel and the interface. If the burst mode latch is on, a condition code of 10 is set and the I/O operation is terminated.

At I/O counter 3 time, the 'execute control' latch is turned on and the subchannel is tested for operational and busy states.

If the subchannel is not operational, condition code 11 is set; if the subchannel is busy, condition code 10 is set and the operation is terminated.

If the subchannel is not busy, the CAW latch is set in order to fetch the CAW from main storage. The I/O counter advances from three to zero.

CAW-Fetch

- The address in the CAW defines the address of the first CCW to be used.
- The CAW is fetched from fixed location 48 hex in main storage.
- Two C-cycles are needed to fetch the CAW.
- Bits 8 to 31 of the CAW are compressed and stored into the COAR in the subchannel.

Figure 3B-50 shows the CAW format. Bits 0 to 7 and bits 8 to 14 must always be zero.

The CAW latch sets the C-cycle request latch which turns on the 'C-cycle control' and 'C-cycle' latches.

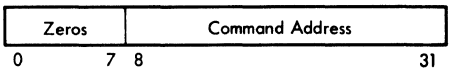


Figure 3B-50. CAW Format

First C-Cycle

The SAR is reset and the fixed address (48 hex) is loaded. The CAW is read out to the SDR. At RC3, the I/O counter is advanced to one; this selects halfword 0 (byte 0 and byte 1 in the SDR) to be gated to the subchannel. B0 and B1 data registers are loaded at WC2 from byte 0 and byte 1 in the SDR (byte 0 is now all zeros).

Second C-Cycle

At the beginning of the next cycle, the fixed address is forced to the SAR and the CAW is again read out to the SDR. At RC3, the I/O counter advances to zero and halfword 1 is gated to the subchannel. Bytes 2 and 3 in the SDR are gated straight into bits 16 to 28 of the COAR. Bits 8 to 14 in the B1 data register are gated through the compress and the output is loaded into bit 14 of the COAR. Bit 15

of the B1 data register is gated directly into bit 15 of the COAR.

The CAW latch is reset by I/O counter 0 during this cycle, resulting in the C-cycle request latch being turned off.

The 'program check' latch is set if any of the following conditions occur during these two cycles:

SDR parity error (also sets the channel control check and machine check).
Bits 29 to 31 are not zero.
Bits 0 to 7 are not zero.

The CCW latch is set during the last C-cycle in order to fetch the CCW.

CCW-Fetch

- CCW is loaded into the subchannel.
- Four I/O cycles are needed to fetch the CCW.
- The command code and bits 8 to 15 of the data address are loaded into B0 and B1 data registers respectively on the first cycle.
- The data address is loaded into the DAR on the second cycle.
- The count is loaded into the count register.
- The flag bits are loaded during the fourth I/O cycle.

Since CCW is a double word, four I/O cycles are needed to load the information into the subchannel. The format of the CCW is shown in Figure 3B-51. The CCW latch (set during the last CCW-fetch cycle) sets the I/O cycle request latch which turns on the 'I/O cycle control' and 'I/O cycle' latches.

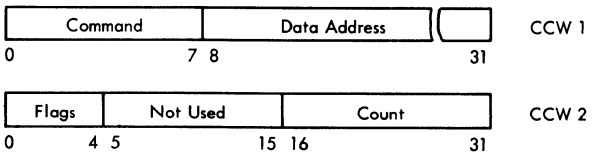


Figure 3B-51. CCW Format

First I/O Cycle

At the beginning of the first I/O cycle, the contents of the COAR are gated to the SAR and CCW 1 is read out to the SDR. At RC3, the I/O counter is advanced to one, and halfword 0 is selected to the subchannel. The command is gated to B0 data register and bits 8 to 15 of the data address are loaded into B1 data register.

During this cycle the 'program check' latch is set if:

A TIC command is detected.
Bits 4 to 7 are zero.
A parity error is detected in the SDR.

Second I/O Cycle

On the second I/O cycle, CCW 1 is again read out to the SDR. At RC3, the I/O counter is advanced to two, and halfword 1 is selected to the subchannel. Bits 15 to 31 of CCW 1 are gated directly into bits 15 to 31 of the DAR. Bits 8 to 14 of the B1 data registers are gated through the compress and loaded into bit 14 of the DAR. Bit 15 of the B1 data register is gated directly into bit 15 of the DAR.

The contents of the B0 data register are decoded and loaded into the Op register. At the same time, the B2 buffer register is loaded from the B0 data register. The command is now stored in the B2 buffer register until it is sent to the control unit during initial selection. The 'program check' latch is set during this cycle if a parity error is detected in the SDR.

Third I/O Cycle

At the beginning of this cycle the content of the COAR is gated to the SAR. In order to read out CCW 2, bit 29 is forced from the I/O counter. At RC3, the I/O counter is advanced to three, and halfword 1 is selected to the subchannel. The count is now loaded into the count register. Halfword 1 is selected during this cycle so that the count register may be checked for zero after it is loaded. This checking is done during the next I/O cycle.

The 'program check' latch is set during this cycle if:

A parity check is detected in SDR.
Bits 5 to 7 of CCW 2 are not zeros.

Fourth I/O Cycle

The SAR is loaded as during the third I/O cycle and CCW 2 is again read out to the SDR. The I/O counter advances to zero, and halfword 0 is selected to the subchannel. B0 and B1 data registers are loaded and bits 0 to 4 in B0 data register are gated into the flag bits.

The CCW latch is reset at I/O counter 0 time, resulting in resetting the 'I/O cycle request' latch. The value in the COAR is increased by eight in order to define the address of the next CCW.

The 'program check' latch is set during this cycle if the content of the count register is zero or if a parity error in the SDR is detected.

CCW-fetch is now complete and the 'program check' latch is analyzed. If the 'program check' latch is off, the 'CPU IF request' latch is set in order to continue with initial selection.

If the 'program check' latch is on, the 'CSW store' latch is set. The status part of the CSW is stored and a condition code of 01 is set, as described in "Storing the CSW" under the heading "Start I/O and Halt I/O."

Initial Selection

- Initial selection is the operation used to select a control unit and device and is initiated from the subchannel.
- Initial selection is used at the end of CCW-fetch for start I/O and command-chaining operations.
- Initial selection is also used for test I/O and for a device-end interrupt.
- An address is sent to bus out in order to select the control unit and device.
- The control unit responds with an address on bus in.
- The command is sent to bus out.
- The control unit sends a status byte to bus in.
- The status byte is analyzed.

The channel is connected to the control units by means of the standard interface, a description of which is contained in IBM System/360 I/O Interface Channel to Control Unit, OEMI, Form A22-6843.

At WC4 during the last CCW-fetch cycle, the 'CPU IF request' latch is set. If the interface is at this time working in burst mode, and the 'burst mode' latch is on, a condition code of 10 is set and the operation is terminated.

If the burst mode latch is off, the 'IF requested and free' latch is set as soon as the interface is free. (See subsequent Figure 3B-65.) At the same time, the 'CPU select IF' latch is turned on. Bits 27 to 31 are now gated from the B register to bits 3 to 7 of bus out, and bit 0 of bus out is forced from the 'IF requested and free' latch. Bits 1 and 2 of bus out are gated from the IF subchannel select register (address bits 1 and 2). The 'address out' tag is gated 600 nanoseconds after the 'IF requested and free' latch is set, and the 'select out' tag is raised 750 nanoseconds after the 'address out' tag.

The first-cycle latch is set in order to indicate that this operation is an initial-selection sequence.

The control unit that matches the address on bus out is now selected. If no control unit is selected, 'select in' is raised. This means that the addressed control unit is not connected; a condition code of 11 is set and the operation is terminated.

The selected control unit may respond with either 'status in' or 'operational in'.

The 'status in' tag is raised if the selected control unit is busy at this time. A status byte is gated to bus in, and the delay-line drive is started. At tag delay 1, the B2 buffer register is reset and, at tag delay 2, the register is loaded from bus in. The status byte is now analyzed, resulting in the status portion of the CSW being stored and a condition code of 01 being set.

The 'operational in' tag is raised if the selected control unit is not busy. 'Address out' is reset by 'operational in'. The fall of 'address out' is detected by the control unit and an address is gated to bus in with the 'address in' tag. 'Address in' starts the delay-line drive. At tag delay 2, the DEVAR is reset and, at tag delay 3, bits 3 to 7 of bus in are loaded into the DEVAR. At IF delay, bus in and bus out are compared in order to check that the correct control unit and device are selected. If the addresses do not match, the IF control check is set, resulting in a machine check. Bad parity on bus in also sets the IF control check.

At tag delay 4, the command in the B2 buffer register is gated to bus out, and the 'command out' tag is raised at the end of delay line 3 (250 nanoseconds later). A parity error on bus out sets the channel control check.

When 'command out' is detected in the control unit, 'address in' is reset which turns off the 'command out' latch in the subchannel. At the fall of 'command out' (detected by the control unit), a status byte is gated to bus in and the 'status in' tag is raised.

The 'status in' tag turns on the 'accept status' latch in the subchannel and the delay-line drive is started. B2 buffer register is reset at tag delay 1 and loaded from bus in at tag delay 2. At the same time, the Status Analysis (STAN) latch is set in order to analyze the status byte. The 'service out' tag is raised at IF delay.

When the 'service out' tag appears at the control unit, the 'status in' tag is reset. This resets the delay-line drive in the subchannel. The 'service out' latch stays on until the delay-line drive is timed out. The initial-selection sequence is now complete.

Status Analysis during Initial Selection

Read this section in conjunction with FEMD Figure 6140.

Two conditions may exist during initial selection: control unit busy and control unit not busy.

Control Unit Busy

This analysis results in setting the CSW latch because the 'channel end' bit is not on. At this time, it is not necessary to gate 'service out'.

Control Unit Not Busy

The three types of status byte are handled as follows.

Status Equals Zero: If the status byte is zero (all bits on bus in are zero) a condition code of 00 is set in the PSW.

Channel-End Bit On: Channel end is presented in the status during initial selection if an immediate operation (such as 'rewind tape') is executed. If command chaining is not indicated in the subchannel, the status portion of the CSW is stored and a condition code of 01 is set in the PSW. If command chaining is indicated and no subchannel error or halt condition is indicated, a condition code of 00 is set in the PSW. 'Suppress out', followed by 'service out' 250 nanoseconds later, is raised in order to indicate command chaining to the device.

Any Other Bits On: When a status byte is being presented during initial selection and any bit (other than 'channel end') is on, the status portion of the CSW is stored and a condition code of 01 is set in the PSW.

Release CPU and Commence I/O Operation

Any condition code setting releases the CPU from the channel and resets the 'execute control' and 'channel decode' latches.

When a zero status is analyzed and the subchannel is set up for a write operation, the I/O cycle request latch is set in order to have data ready in B2 data register when the first 'service in' appears.

Start I/O execution is always ended when a condition code of 00 is set. A condition code of 00 means that the subchannel now controls the rest of the I/O operation.

After the completion of initial selection, the control unit works in either byte mode or burst mode.

In byte mode, the control unit drops 'operational in' and another control unit and subchannel may now use the interface. Each time the control unit needs to communciate with the subchannel, 'request in' is raised. In burst mode, 'operational in' stays up until the entire transfer of data between the selected subchannel and device is finished. When the control unit requires to communicate with the subchannel, the 'service in' is raised.

Data Service

- Data service is initiated from the control unit.
- The HSMPX channel can perform read-forward, read-backward and write operations.
- The data address at the begining of a read or write operation may specify a byte, halfword or word boundary.
- A maximum of two bytes can be transferred at one time between the subchannel and the SDR.

A data transfer between the subchannel and the control unit is always initiated from the control unit. Each time a control unit has a byte of information for a read operation or needs a byte of information for a write operation, a request is sent to the channel.

In burst mode, the control unit and the subchannel are always connected, and the control unit raises only 'service in'. In byte mode, the control unit that wishes to transfer a byte of information, raises 'request in' in order to select the required subchannel. The sequences which are followed when the control unit has raised either 'service in' or 'request in' are described subsequently in "Service-In Sequence."

The three types of data transfer in the subchannel are:

Read forward
Read backward
Write.

For a write operation, the data path is from main storage via the subchannel, interface and control unit to the device. For a read operation, the data path is reversed.

At the beginning of a read or write operation, the data address in the DAR may specify a byte, halfword or word boundary in main storage. There are therefore four possible starting addresses within any word as shown in the following table.

Bit 30 of DAR	Bit 31 of DAR	Boundary	Byte in SDR
0	0	Word	Byte 0
0	1	Byte	Byte 1
1	0	Halfword	Byte 2
1	1	Byte	Byte 3

As bits 30 and 31 of the DAR are not used to address main storage, a word confined within a word boundary is always read out to the SDR. Bits 30 and 31 are used to select the data path between the B2 buffer register and the SDR.

Because each subchannel has two data registers, only two bytes can be transferred between the SDR and the subchannel at one time.

Read Operation

- The data path for a read operation is from the device via the control unit, B2 buffer register and B0, B1 data registers to main storage.
- Bit 31 of the DAR selects the data path from B2 buffer register to the data registers.
- Bit 30 of the DAR selects the data path from the data register to SDR.
- An I/O cycle is requested each time B2 is gated to B1 for a read forward.
- An I/O cycle is requested each time B2 is gated to B0 for a read backward.
- The 'I/O cycle extend' latch is used to indicate that the B2 buffer register is empty for a read operation.

Figure 3B-52 shows a simplified data flow for a read operation. During 'service in' for a read operation, a byte is gated to the B2 buffer register when this is empty. B2 buffer register may now be gated to either the B0 or B1 data registers. The data registers may then be gated to either halfword 0 or halfword 1 in the SDR. The data may be transferred either for read forward or read backward.

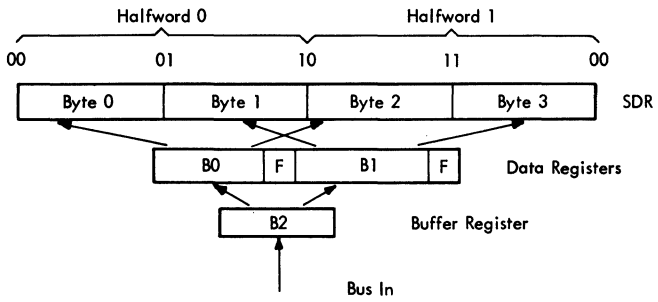


Figure 3B-52. Data Flow (Read)

Read Forward: The data address defines the high-order byte in main storage, and the value held in the DAR is increased by two during each I/O cycle. Bit 31 of the DAR is always changed each time a data byte is gated to the B2 buffer register.

The 'service in' sequence is started only when B2 buffer register is empty. To check if B2 register is empty, the I/O cycle extend latch and bit 31 of the DAR are tested; the on states of both the latch and bit 31 indicate the B2 buffer register is not empty. The logic circuit is shown in Figure 3B-53.

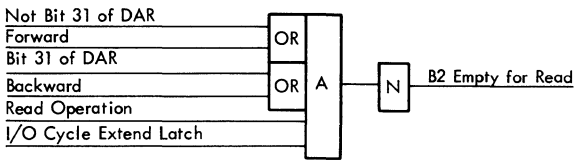


Figure 3B-53. B2 Empty for Read

If the starting address is confined within a word boundary (bits 30 and 31 of DAR are off), the first byte loaded in the B2 buffer register must be gated to the B0 data register and later to byte 0 of the SDR. Because bit 31 of the DAR was changed when B2 buffer register was loaded, the following rule is applied:

B2 is gated to B0 if bit 31 of DAR is on, and to B1 if bit 31 of DAR is off.

B2 buffer register is transferred to the data registers during the 'service in' sequence if the I/O cycle extend latch is off. The on state of this latch indicates that both B0 and B1 data registers are full; the contents of B2 buffer register are (in this case) transferred to the data registers when the I/O cycle is taken. The gating of B2 buffer register to the data registers is shown in Figure 3B-54.

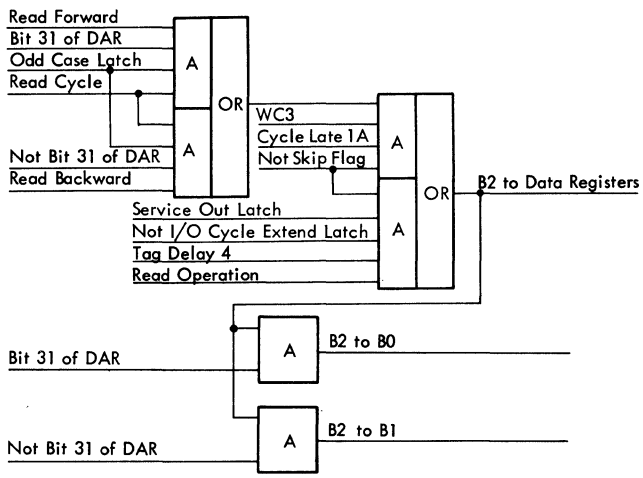


Figure 3B-54. Gating of B2 Buffer Register to Data Register

The value in the DAR is increased at the end of each I/O cycle, and the following rule is then applied:

B0 and B1 data registers are loaded into bytes 0 and 1 of SDR if bit 30 of DAR is off, and to bytes 2 and 3 if bit 31 of DAR is on.

In order to load B0 and B1 data registers to the SDR, the associated flag bits must also be on (see "Byte Control").

During a read operation, the B0 and B1 data registers always try to keep B2 buffer register empty and an I/O cycle is requested as soon as B1 data register is full. The I/O cycle request latch is set each time B2 register is gated to B1 register (see Figure 3B-55).

The I/O cycle is taken as soon as the priority for the channel and subchannel is established. During this cycle the content of the DAR (not bits 30 and 31) is loaded into the SAR and one word is gated to the SDR from main storage. The bytes loaded into the SDR depend on the byte control. At the beginning of a read operation either one or two bytes may be loaded into the SDR from the subchannel, depending upon the state of bit 31 of DAR. The following table shows the loading of SDR for the four different starting addresses during the first I/O cycle:

Data Address Bit 30	Data Address Bit 31	Byte Control	Bytes Loaded from Storage	Bytes Loaded from Channel
0	0	0.1	2.3	0.1
0	1	1	0.2.3	1
1	0	2.3	0.1	2.3
1	1	3	0.1.2	3

During the subsequent I/O cycles, two bytes are always loaded into the SDR from storage and two from the channel, provided that the count is not zero.

At the end of the operation, the loading of the SDR during the last I/O cycle depends on the ending address, as shown in the following table:

Data Address Bit 30	Data Address Bit 31	Byte Control	Bytes Loaded from Storage	Bytes Loaded from Channel
0	0	0	1.2.3	0
0	1	0.1	2.3	0.1
1	0	2	0.1.3	2
1	1	2.3	0.1	2.3

Read Backward: For a read-backward operation, the data address defines the low-order byte in main storage. To place the data bytes in the same order as for a read-forward operation, the value held in the DAR is reduced during each I/O cycle.

The I/O cycle extend latch is used during read backward to test if B2 buffer register is empty. The register is not empty if the latch is on and if bit 31 of the DAR is off (see Figure 3B-53).

If the starting data address for a read-backward operation is on a word boundary, the first byte from bus in must be loaded into byte 0 of the SDR (see Figure 3B-56). The following rules are applied:

B2 is gated to B0 if bit 31 of DAR is off, and to B1 if bit 31 of DAR is on.

B0 and B1 data registers are loaded into bytes 0 and 1 of SDR if bit 30 of DAR is off, and into bytes 2 and 3 if bit 30 of DAR is on.

This rule assumes that both flag bits are on.

From Figure 3B-56, it can be seen that the I/O cycle request latch must be set when the first byte

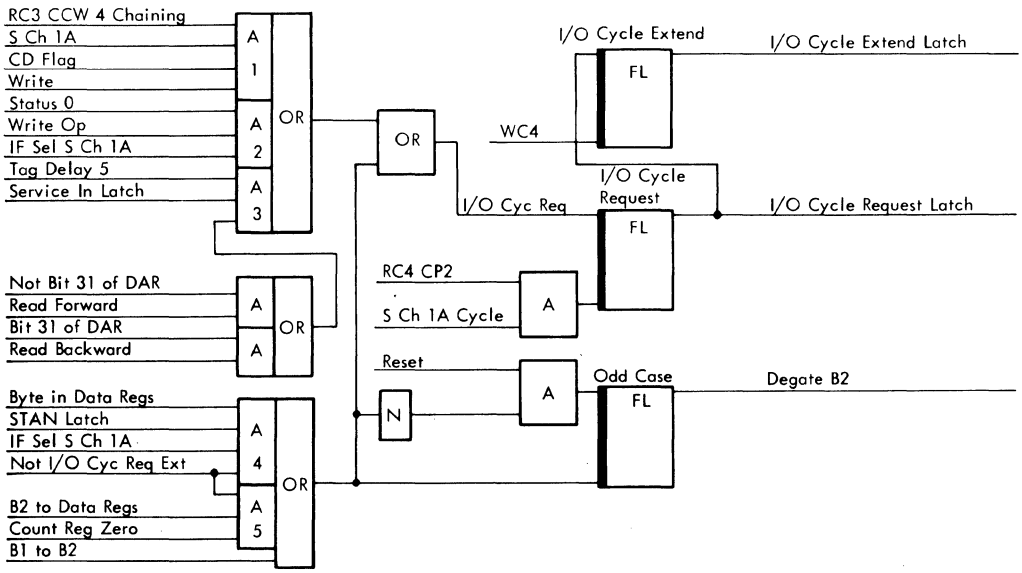


Figure 3B-55. I/O Cycle Request and Extend Latches

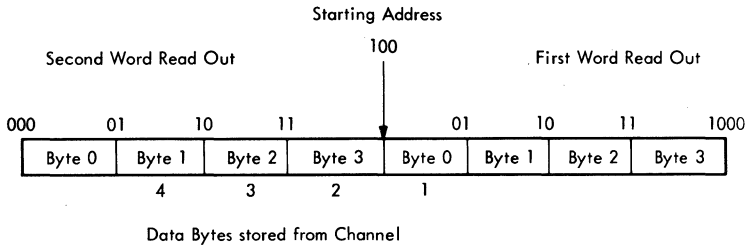


Figure 3B-56. Example of Even Starting Address (Read Backward)

has been loaded from B2 to B0. Subsequently, the following rule applies:

An I/O cycle is requested each time a byte of data has been loaded from B2 to B1 (Figure 3B-55).

The loading of the SDR, during the first I/O cycle, for the four different starting addresses is shown in the following table:

Data Address Bit 30	Data Address Bit 31	Byte Control	Bytes Loaded from Storage	Bytes Loaded from Channel
0	0	0	1.2.3	0
0	1	0.1	2.3	0.1
1	0	2	0.1.3	2
1	1	2.3	0.1	2.3

During the subsequent I/O cycles, two bytes are always regenerated and two bytes are loaded into the SDR from the channel, provided that the count is not zero.

The loading of SDR during the last I/O cycle depends on whether the last byte is located on a byte, halfword or word boundary in main storage, as shown in the following table:

Data Address Bit 30	Data Address Bit 31	Byte Control	Bytes Loaded from Storage	Bytes Loaded from Channel
0	0	0.1	2.3	0.1
0	1	1	0.2.3	1
1	0	2.3	0.1	2.3
1	1	3	0.1.2	3

End of Read Operation: When an end operation is signalled to the subchannel, the data registers and the B2 buffer register are analyzed to determine if data is stored in the subchannel. If data is stored, the I/O cycle request latch is set to store the remaining data bytes. The 'end' signal is generated, either from the channel (when the count goes to zero) or from the control unit, by a status byte.

When the count goes to zero during a 'service in' sequence, three different conditions may occur:

1. Both data registers are empty. The last byte is gated to the data registers as explained previously for read forward or read backward. As seen in Figure 3B-55 (AND block 5), the I/O cycle request latch is set if B2 buffer register is gated to either of the data registers. When the I/O cycle is taken, the data byte in the data register defined by the flag bit is stored in main storage.

2. One byte is stored in the data register. The data byte in B2 buffer register is gated to the empty data register and the I/O cycle request latch is set as explained previously.

3. Both data registers are full and the I/O cycle request latch is on. B2 buffer register is gated to the data registers during the previously-requested I/O cycle. When the I/O cycle extend latch goes off, the I/O cycle request latch is set and the 'odd case' latch is turned off; see Figures 3B-55 (AND block 5) and 3B-57. During the last I/O cycle, the data byte in the data register is stored in main storage.

When 'status in' appears in the subchannel and there is still data in the data registers, the I/O cycle request latch is set; see Figure 3B-55 (AND block 4). This may occur if 'status in' appears before the count is zero.

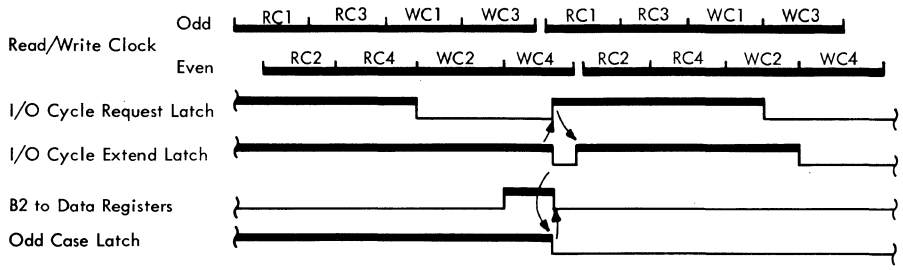


Figure 3B-57. End Read (Three Bytes of Data in Subchannel)

Write Operation

- The data path for a write operation is from main storage via the data registers, B2 buffer register and control unit to the device.
- Bit 30 of the DAR controls the data path from the SDR to the data registers.
- Bit 31 of the DAR controls the data path from the data registers to the B2 buffer register.
- An I/O cycle is requested each time B1 data register is gated to B2 buffer register.
- At the beginning of a write operation, data is fetched in order to have B2 full when the first 'service in' appears.

A simplified data flow for a write operation is shown in Figure 3B-58. During an I/O cycle, two bytes are transferred from either halfword 0 or halfword 1 in the SDR to the B0 and B1 data registers in the subchannel; this data flow is under the control of bit 30 of the DAR. One byte is then loaded into the B2 buffer register from either B0 or B1 data registers, depending on the state of bit 31 of DAR.

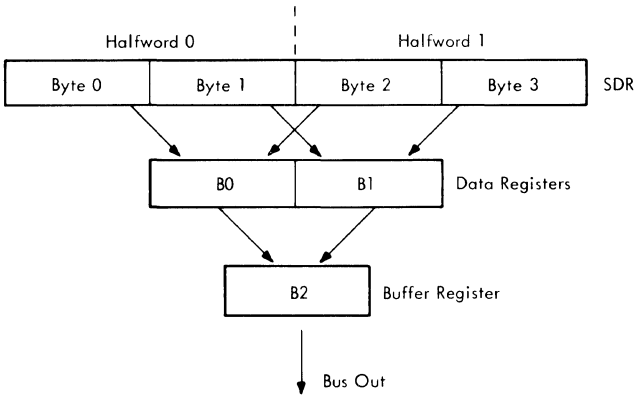


Figure 3B-58. Data Flow (Write)

A word defined by the DAR is always read out to the SDR and all bytes are regenerated.

The subchannel always tries to keep a byte of data in the B2 buffer register and as soon as the data registers are empty, an I/O cycle is requested; Figure 3B-58 shows that the data registers are empty when B1 is gated to B2. The I/O cycle request latch is set each time B1 is gated to B2.

The value in the DAR is increased by two during each I/O cycle and bit 31 of the DAR is changed each time a byte of data is gated to bus out.

The data service for a write operation consists of data-fetch for write, and data service during a 'service in' sequence. These are detailed subsequently.

Data-Fetch for Write

- Data is fetched at the beginning of a write operation in order to have data ready in the subchannel when 'service in' appears.
- The I/O cycle request latch is set when the status byte is equal to zero during initial selection for start I/O and command chaining.
- The I/O cycle request latch is set during the last CCW-fetch cycle for a data-chaining operation.
- An even starting-data address results in a request for one I/O cycle.
- An odd starting-data address results in a request for two I/O cycles.

For a write operation, data is normally requested when the B1 data register is gated to the B2 buffer register during the 'service in' sequence, but the 'service in' sequence cannot start before the B2 buffer register is full. To start the data transfer for write, the I/O cycle request latch must therefore be set before 'service in' appears.

The I/O cycle request latch is set when the status is analyzed and found equal to zero during the initial-selection sequence for start I/O and command chaining. For data chaining, however, the I/O cycle request latch is set during the last CCW-fetch cycle. See Figure 3B-55 (AND blocks 1 and 2).

Since the data address can be confined within a byte boundary in storage, one or two I/O cycles may be requested, depending on the position of the byte within the word.

Even Starting Data Address

If the data address is even (that is, when bit 31 of DAR is off), either byte 0 or byte 2 in the SDR must be the first byte to be gated to the B2 buffer register (refer to Figure 3B-58).

During the first I/O cycle, the DAR is gated to the SAR and four bytes are loaded into the SDR. Bit 30 of the DAR selects either bytes 0 and 1, or bytes 2 and 3, for gating to the B0 and B1 data registers. If bit 31 of the DAR is off, the contents of B0 data register are gated to the B2 buffer register. When the first 'service in' appears, one byte is ready in the B2 buffer register and another byte is ready in the B1 data register.

Odd Starting Data Address

If the data address is odd (that is, when bit 31 of DAR is on), byte 1 or byte 3 in the SDR must be the first data byte to be gated to the B2 buffer register depending on the state of bit 30 of the DAR.

Assume that bit 30 of the DAR is off. During the first I/O cycle, bytes 0 and 1 in the SDR are gated to the B0 and B1 data registers. If bit 31 of the DAR is on, the contents of the B1 data register are gated to the B2 buffer register. 'B1 to B2' means that the data registers are empty and another I/O cycle is requested. During this second I/O cycle, the data registers are loaded from bytes 2 and 3 in the SDR. When the 'service in' now appears, three bytes of data are available in the subchannel.

The 'odd case' latch (see Figure 3B-55) gates B1 data register to B2 buffer register during the first cycle. This latch is set when the subchannel is reset, and is reset when B1 register is gated to B2 register. The timing of the I/O cycle request latch and the 'odd case' latch is shown in Figure 3B-59.

Figure 3B-59. Data Fetch for Write (Odd Address)

Service-In Sequence

- The I/O cycle extend latch indicates that the B2 buffer register is full for a write operation.
- Data is transferred from the data registers to the B2 buffer register at tag delay time if the I/O cycle extend latch is off.
- Data is transferred from the data registers to the B2 buffer register during the I/O cycle if bit 31 of the DAR is off.
- If bit 31 of the DAR is on, B1 data register is gated to B2 buffer register.
- If bit 31 of the DAR is off, B0 data register is gated to B2 buffer register.

When the first 'service in' appears in the HSMPX channel and the selected subchannel is set for a write operation, the 'service in' sequence is started only if B2 buffer register is full. To check if this register is full or not, the I/O cycle extend latch is tested; if the latch is on, B0 and B1 data registers are empty and, if bit 31 of the DAR is off at the same time, B2 buffer register also is empty. This test holds good for all cases except when the starting address is odd (that is, when bit 31 of the DAR is on). To correct for this case, the 'odd case' latch is used to indicate when B2 buffer register is empty; the logic for this is shown in Figure 3B-60. The 'odd case' latch is reset at the same time as B1 register is gated to B2 register.

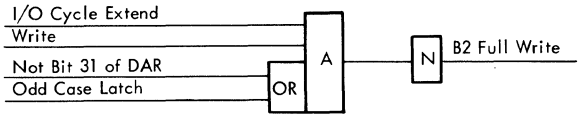


Figure 3B-60. B2 Full for Write

When the 'service in' sequence is started and the byte in the B2 buffer register has been gated to bus out, a new byte of data is loaded immediately into this register if the I/O cycle extend latch is off.

If the I/O cycle extend latch is on, however, the B2 buffer register is loaded from the data registers during the I/O cycle, the off state of bit 31 of the DAR gates B0 register to B2 register and the on state of the bit gates B1 register to B2. Figure 3B-61 shows the logic of gating the data registers to the B2 buffer register.

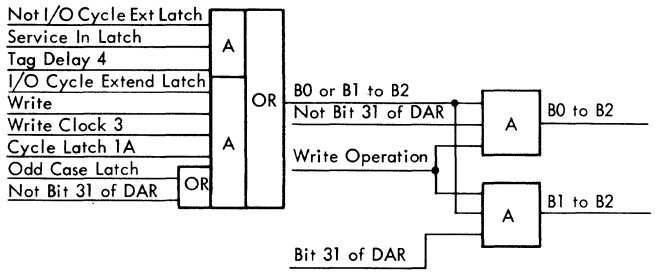


Figure 3B-61. Gating of Data Registers to B2 Buffer Register

Request-In Sequence for a Service-In

- A control unit sends 'request in' in order to select a subchannel.
- 'Request in' is sent only when the device is working in byte mode.
- 'Request in' is raised each time the device needs to communicate with a subchannel.
- The subchannel is always busy when a 'request in' is raised for service.

Since up to four subchannels may be connected to the standard interface, the device must select the correct subchannel before data can be transferred. In burst mode, the device and the subchannel are always connected; there is, therefore, no need to send 'request in'. In byte mode, 'request in' is raised each time the device needs to communicate with the subchannel.

A 'request in' may be raised at any time from any control unit, and more than one control unit may raise 'request in' at the same time. When the subchannel detects 'request in', the select out tag is raised as soon as the interface is free. If other control units have also raised 'request in', the control unit with the highest physical priority is selected.

The selected control unit raises 'operational in', which then drops the request in tag. An address is also gated to bus in, followed by 'address in'.

Bits 1 and 2 on bus in are gated to the subchannel select register in order to select the correct subchannel. An 'address in' tag detected in the channel also starts the delay-line drive and resets the select out latch.

At delay line 2, the DEVAR control latch is set to prevent the DEVAR from being reset and reloaded. The parity on bus in is checked. If bad parity is detected, the IF control check latch is set.

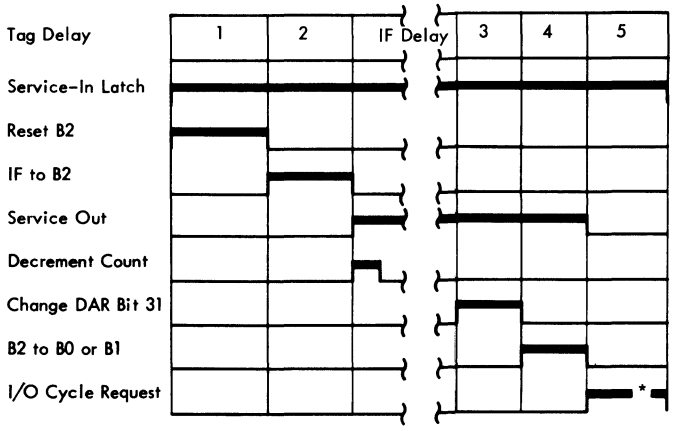
'Command out' is gated at tag delay 5, and zeros are gated to bus out. The control unit responds by dropping 'address in' and, when the delay-line drive has timed out, 'command out' is dropped.

When the fall of 'command out' is detected, the control unit raises 'service in'. At this time, the selection of the subchannel from the interface is complete and the 'service in' sequence continues with the required read or write operation.

Service-In for a Read Operation

- 'Service in' is raised if the control unit has a byte of data to send to the subchannel.
- The delay-line drive is started only when the B2 buffer register is empty.
- The 'service in' latch is set by the delay-line drive.
- The B2 buffer register is reset at tag delay 1.
- The interface is gated to the B2 buffer register at tag delay 2.
- 'Service out' is gated at IF delay time.
- The value held in the count register is reduced by one at IF delay time.
- Bit 31 of the DAR is changed at tag delay 3.
- The B2 buffer register is gated to either B0 or B1 data register at tag delay 4 if the I/O cycle extend latch is off.
- The I/O cycle request latch is set if bit 31 of the DAR is off for read forward.
- The I/O cycle request latch is set if bit 31 of the DAR is on for read backward.

The 'service-in' sequence for a read operation is shown in Figure 3B-62.



* If bit 31 of DAR On for Read Backward; if bit 31 of DAR Off for Read Forward

Figure 3B-62. Service-In (Read)

When 'service in' is raised and the subchannel is set up for a read operation, the delay-line drive is turned on only if the B2 buffer register is empty; the data byte from the previous 'service in' is thus transferred to either the B0 or B1 data register. At the beginning of a read operation, B2 buffer register is always empty and the delay line drive is set immediately. If the B2 buffer register is full, 'service in' stays up until the register is empty.

The delay-line drive sets the service in latch and the B2 buffer register is reset at tag delay 1. At tag delay 2, the data byte on the interface is gated to the B2 buffer register. The value held in the count register is reduced by one and 'service out' is raised at IF delay time. At the same time, the parity on bus in is checked and the channel data check is set if the parity is bad.

When 'service out' is detected by the control, 'service in' drops and the delay-line drive trigger is again turned off.

Bit 31 of the DAR is changed at tag delay 3. On the next tag delay (4), the B2 buffer register is gated to B0 or B1 data register if the I/O cycle extend latch is off. (The on state of this latch means that the B0 and B1 data registers are not yet empty.) When bit 31 is on, B2 buffer register is gated to the B0 data register; when bit 31 is off, B2 register is gated to B1 data register. If the I/O cycle extend latch is on at tag delay 4, the B2 buffer register is gated to the B0 and B1 data registers at WC4 during the first I/O data cycle.

The service out latch is reset at the end of tag delay 4.

At tag delay 5, the I/O cycle request latch is set if bit 31 of the DAR is off for read forward or if the bit is on for read backward.

When the requested I/O data cycle is taken, B0 data register is gated to bytes 0 and 2 in the SDR and B1 data register is gated to bytes 1 and 3 in the SDR. If bit 30 of the DAR is off and both flag bits in the B0 and B1 data registers are on, bytes 0 and 1 are loaded. When bit 30 of the DAR is on, bytes 2 and 3 are loaded into the SDR provided that both flags in B0 and B1 data registers are on. (Refer to Figure 3B-37.)

Termination of a Read Data Service

The termination of a read data service is governed by one of three different conditions:

1. When the number of bytes in the record is equal to the contents of the count register. During the last 'service in', the count register goes to zero and 'service out' is raised in response to 'service in'. A count of zero also causes the I/O cycle request latch to be turned on. When the control unit detects that no more information is available at the device, 'request in' is raised and a status byte is sent to the subchannel that has the 'channel end' bit on. The analysis of this status byte is described subsequently in "End Procedure" under the heading "Status Analysis (Status-In Sequence)."

2. When the number of bytes in the record is greater than the contents of the count register. During the 'service in' sequence in which the count register goes to zero, the I/O cycle request latch is set and 'service out' is raised as a response to the control unit. The chain-data flag is then tested.

If the CD flag is on, the data-chaining routine is entered as soon as all I/O cycle requests are taken. During the next 'service in' sequence, the program check and the channel data check are tested. If either of these is on, 'command out' is gated to the control unit which then generates the channel-end condition. If no checks are detected, however, the read operation proceeds as normal until the count again goes to zero or a 'status in' is presented.

If the CD flag is off, no action is taken until the next 'service in' appears. The command out tag is then raised and the incorrect length (IL) latch is set. 'Command out' is detected by the control unit which then generates the channel-end operation; this operation is presented to the subchannel during a later 'request in'.

3. When the number of bytes in the record is less than the contents of the count register. In this case, a 'status in' is presented to the subchannel before the count register is zero. The analysis of this status byte sets the 'incorrect length record' flag provided that the SLI flag is not on.

Service-In for a Write Operation

- 'Service in' is raised if the control unit requires a byte of data.
- The delay-line drive is set only if a byte of data is stored in the B2 buffer register.
- The delay-line drive sets the 'service in' latch.
- The contents of B2 buffer register are gated to bus out at tag delay 1.
- 'Service out' is raised at IF delay time.
- The value held in the count register is reduced by one at IF delay time.
- B2 buffer register is reset at tag delay 3.
- DAR bit 31 is changed at tag delay 3.

- B0 or B1 data register is gated to B2 buffer register during tag delay 4 if no I/O data cycles are requested.

- The I/O cycle request latch is set if bit 31 of the DAR is off.

The 'service in' sequence for a write operation is shown in Figure 3B-63.

When 'service in' is raised and the subchannel is set up for a write operation, the delay-line drive is started only if the B2 buffer register is full. If the register is full, a byte of data is ready to be transferred to bus out. At the beginning of a write operation, one or (in some cases) two storage cycles are taken in order to have data ready in the subchannel before the first 'request in' appears (see "Data Fetch for Write").

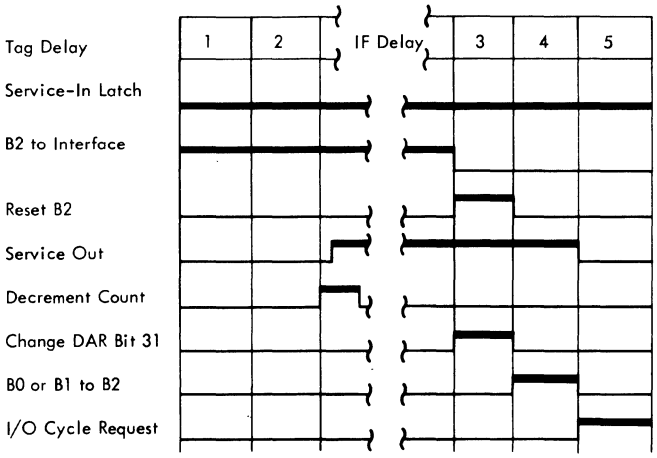


Figure 3B-63. Service-In (Write)

The on state of the delay-line drive trigger turns on the 'service in' latch. At tag delay 1, the B2 buffer register is gated to bus out. The count is reduced and 'service out' is raised at IF delay time. B2 buffer register is reset and DAR bit 31 is incremented at tag delay 3. The I/O cycle request latch is now tested. If the latch is off, either the B0 or the B1 data register is gated at tag delay 4 to the B2 buffer register, depending on bit 31 of the DAR. If bit 31 is on, B0 is gated to B2; if bit 31 is off, B1 is gated to B2. Should bit 31 be off, the I/O cycle request latch is set at tag delay 5 to fetch two more data bytes. The value in the DAR is increased by two for each storage cycle.

Termination of a Write Data Service

The termination of a write data service is governed by one of three different conditions:

1. When the contents of the count register are equal to the maximum number of bytes that can be written on a device (example: 80 bytes for IBM 1442 Card Read Punch). When the 'service in' for the last byte is handled in the subchannel, the count goes to zero and 'service out' is raised in response to 'service in'. The device detects the channel-end condition and a status byte is sent to the subchannel for further analysis.

2. When the contents of the count register are greater than the maximum number of bytes that can be written on one record. In this case, a status byte is presented to the subchannel before the count register has reached zero. The analysis of this status terminates the write operation.

3. When the contents of the count register are less than the maximum number of bytes that can be written in one record. During the service in sequence in which the count register goes to two, 'service out' is raised and the CD flag is tested.

If the CD flag is on, the data-chaining routine is entered. During the following 'service in', the program and channel control checks are tested; the command out tag is raised and zeros are gated to bus out. The control unit then generates the channel-end condition, which is presented to the subchannel during a later 'status in' sequence. If no checks are detected, the write operation proceeds until the count again goes to zero or a 'status in' is presented.

If the CD flag is off, no action is taken until the next 'service in' sequence. The command out tag is then raised and the IL indicator is set. A channel-end condition is presented to the subchannel in a later 'status in' sequence.

End Procedure

- A status byte is sent to the channel in order to signal the end condition.

- 'Channel end' is generated by the control unit when the transfer of data is complete.

- 'Device end' is generated when the device has reached its normal ending point.

- 'Channel end' and 'device end' may occur at the same time for some devices.

- The end condition is signalled to the CPU by the interrupt request latch.

- Each subchannel has one interrupt request latch.

- The interrupt request latches of all four subchannels may be on at one time.

An end condition is always signalled to the channel by a status byte. This status byte is generated in the control unit if either the channel or the device has reached its ending point.

A channel-end condition is generated either when the last byte of information is transferred or when the channel signals the control unit that the count has gone to zero. A device-end condition, however, is detected by the control unit when the device has reached its ending point.

Whenever the control unit has detected 'channel end' or 'device end', the channel is requested in order to present the status byte to the correct subchannel. The status byte is then analyzed in the subchannel and the interrupt request latch may be set. If any subchannel interrupt request latch is on, the I/O interrupt request latch is set and the interrupt is taken when allowed by the CPU.

Request-In for Status-In

- 'Request in' is raised if the control unit has a status byte to present to the channel.
- The subchannel is analyzed to determine whether the status byte can be accepted.
- Status bytes with 'channel end' alone or channel-end with device-end are always accepted by the subchannel.
- A status byte with 'device end' alone is always rejected by the subchannel.
- The status byte is analyzed in the subchannel only if the byte is accepted.
- The subchannel interrupt request latch is set for a device-end status byte only if no subchannel interrupt latches are on.

This 'request in' sequence is used only when the subchannel and device are working in byte mode. During this sequence, the subchannel is selected and tested to check if the following status byte may be accepted by the subchannel.

When any control unit has a status byte to present to the HSMPX channel, 'request in' is raised. As soon as the interface is free, the select out tag is raised from the channel in order to select the control unit. If other control units have also raised 'request in', the control unit with the highest physical priority is selected.

The selected control unit raises the operational in tag which then resets 'request in'. The address of the selected control unit and device is then gated to bus in, and 'address in' is raised.

When 'address in' is detected in the channel, the delay-line drive starts as shown in Figure 3B-39, and the select out latch is reset. Bits 1 and 2 on bus in are loaded into the I/O subchannel select register bits 1 and 2. The outputs of these bits are then decoded in order to select one of the four subchannels as shown in the following table.

IF Subchannel Select Register		Subchannel
Bit 1	Bit 2	
0	0	Subchannel A
0	1	Subchannel B
1	0	Subchannel C
1	1	Subchannel D

At tag delay 3, the IF control check is set if the parity on bus in is incorrect. The subchannel is tested in order to set the DEVAR control latch (see Figure 3B-64). This latch controls the reset and loading of the DEVAR. During a 'request in' sequence, the DEVAR control latch is set if the selected subchannel is busy or if any of the subchannel interrupt latches are on. If the latch is on, the DEVAR is prevented from being reset and reloaded.

The on state of the DEVAR control latch indicates that the following status byte may be:

1. A 'channel end'.
2. A 'channel end' and 'device end' together.
3. A 'device end' when the channel-end interrupt is not yet taken.
4. A 'device end' presented by the other control unit not currently working with that subchannel.
5. A 'device end' during command chaining.
6. An attention status type when the subchannel is busy.

The off state of the DEVAR control latch indicates that the following status byte may be either:

1. A device-end status type when the subchannel is free.
2. An attention status type when the subchannel is free.

In these two cases, the DEVAR control latch is also used to set the subchannel interrupt request latch when 'status in' appears later in the sequence.

At tag delay 5, the selected subchannel is analyzed to see if the following status byte can be accepted by the subchannel or if it must be rejected.

If the subchannel accepts the status byte, 'service out' is raised in response to 'status in' and the status byte is stored and analyzed in the subchannel.

If the subchannel rejects the status byte, the byte is stacked back to the control unit by raising 'command out' in response to 'status in'. In this case, the status byte is not analyzed but the subchannel interrupt request latch may be set. The reject status latch is set during the 'request in' sequence

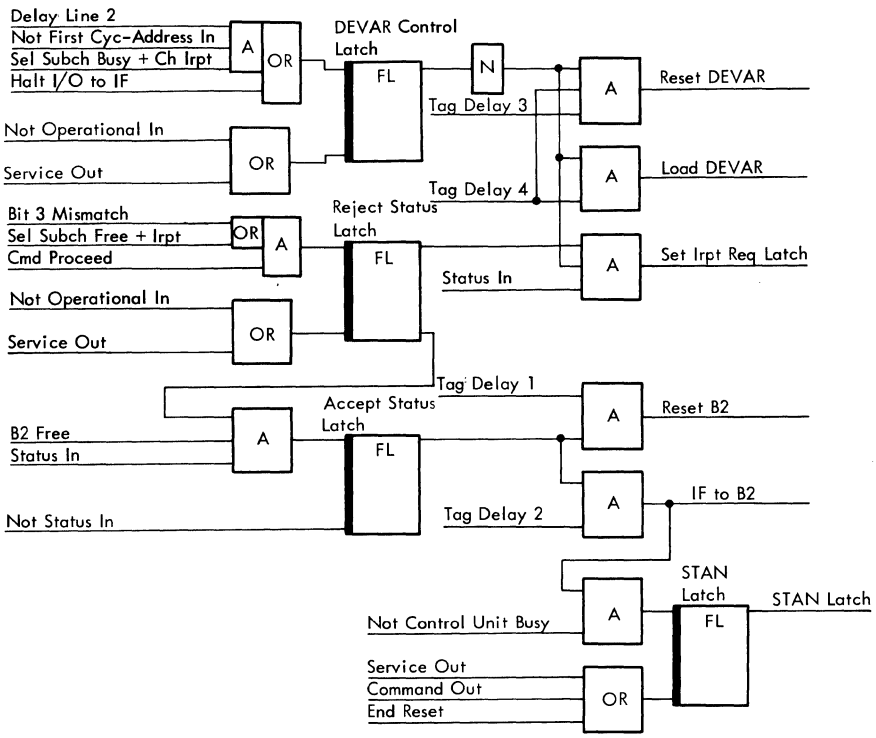


Figure 3B-64. DEVAR Control and Status-In Latches

if the following status byte cannot be accepted by the subchannel during the 'status in' sequence (Figure 3B-64).

Three conditions can set the reject status latch:

1. The selected subchannel is free. The following status byte must be either a device-end or an attention status type. These status bytes are always stacked back to the control unit.

2. The interrupt request latch of the selected subchannel is on. In this case a device-end or an attention status type is presented to the subchannel.

3. Bit 3 does not match. This means that the control unit that has raised 'request in' is not the control unit currently working with the selected subchannel.

Example: Subchannel 1A is connected to control units A and B. Assume that data is transferred between subchannel 1A and control unit in byte mode. If, during this time, a device-end condition is detected by control unit B, 'request in' is raised to present the device-end status byte. Bit 3 of the address on bus in and bit 3 in the DEVAR, however, do not compare and the reject status latch is set. This latch also sets the suppress out latch for the selected subchannel. The suppress out latch stays on until the selected subchannel is reset. Control unit B is prevented from presenting the device-end status until the interrupt generated by control unit A is taken.

At tag delay 5, 'command out' is also raised to indicate to the control unit that it can proceed. The control unit then drops 'address in' which resets the command out latch when the delay-line drive has timed out.

When the fall of 'command out' is detected by the control unit, the status byte is gated to bus in and 'status in' is raised.

Status-In Sequence

When 'status in' appears in the subchannel, the B2 buffer register is tested. If the register is full (the last data byte is not yet transferred for a read operation), 'status in' stays up until the register becomes empty. When this occurs, the delay-line drive is started. Incorrect parity on bus in sets the IF control check.

The sequence which follows depends on the state of the reject status latch:

1. If the reject status latch is on, the status byte is not loaded into the B2 buffer register but is stacked back to the control unit by raising 'command out' at IF delay time. If the DEVAR control latch is off at tag delay 1, the subchannel interrupt request latch is set. This may be a device-end status type which is presented when the selected subchannel is free and if no subchannel interrupt latches are on. (Only one subchannel interrupt request latch may be

on at one time for a device end.) Detection of 'command out' by the control unit stacks the status byte, and the status in and operational in tags are dropped. 'Suppress out' is raised at the time that 'operational in' drops. The command out latch is reset at the end of tag delay 5.

2. If the reject status latch is off, the status byte is accepted by the subchannel, and the 'accept status' latch is set. At tag delay 1, B2 buffer register is reset; at tag delay 2, bus in is loaded and the STAN latch is set in order to analyze the status byte in B2 buffer register.

Status Analysis (Status-In Sequence)

Refer to FEMD, Figures 6139 and 6140. The status byte is analyzed for the following three conditions in the subchannel:

Channel end.
Channel end and device end together
Device end during command chaining.

1. Channel end. If channel-end status is presented to the subchannel during a 'status in' sequence, the CD flag is first tested. If the CD flag is on, it indicates that the record is shorter than indicated by the subchannel and the IL flag is then set. If the CD flag is off, the count register is analyzed; if the count is not zero and the SLI flag is off, the IL flag is then set.

The I/O cycle extend latch is then tested. If it is on, the operation enters a wait state to prevent any further operation from being started until all data bytes are stored in main storage. If the I/O cycle extend latch is off, the CC flag is analyzed.

If the CC flag is off, the 'end' flag and the subchannel interrupt request latch are set. 'Service out' is also raised in response to 'status in', the interrupt is now taken when allowed by the CPU.

If the CC flag is on, the subchannel is tested for errors. Any of the following errors indicate that the subchannel is not fit for chaining:

Incorrect length
Program check
Interface control check
Channel control check
Channel data check.

The on state of the following bits in the status byte also indicates that the subchannel is not fit for chaining:

Bit 0 : Attention
Bit 3 : Busy
Bit 6 : Unit check
Bit 7 : Unit exception.

If the subchannel is unfit for chaining, the 'end' flag and the subchannel interrupt request latch are set and 'service out' is raised.

If no errors are detected, the 'end' flag is set and 'suppress out' is set 250 nanoseconds later. Suppress out is raised to indicate command chaining to the control unit. The subchannel now waits for 'device end' to be presented from the same device.

2. Channel-end and device-end together. If this status type is presented to the subchannel, the analysis is similar to that for a channel-end status type, except that the chaining routine is started immediately. When the 'device end' bit is on, the status modifier bit is also tested and the value held in the COAR is increased by eight if the modifier bit is on. (This bit may be raised during a 'search equal' on disk.)

The CCW request latch is turned on and, when the priority for that subchannel is established, the new CCW is fetched. The interface is not operational ('select out' is inhibited) for other operations until the device is reselected.

3. Device end during command chaining. When 'device end' appears in the subchannel and command chaining is indicated, the 'end' flag is on. To check for an error condition, 'device end' is then tested. If it is off, the subchannel interrupt request latch is set and 'service out' is raised; the CPU may then analyze the status byte when the interrupt is taken.

If the 'device end' bit is on, the subchannel is unfit for chaining because of the same error conditions as those listed previously under item 1, "Channel End."

When the subchannel is not fit for chaining, the subchannel interrupt request latch is set and 'service out' is raised. An attempt is made to set the 'end' flag although, at this time, the flag is already on.

If the subchannel is fit for chaining (no error detected), 'suppress out' and 'service out' are gated, the status modifier bit is tested and the CCW request latch is set as described under item 2, "Channel-end and device-end together."

CHAINING (HSMPX CHANNEL)

The two types of chaining operations are data chaining and command chaining. They are indicated in the subchannel by the CD and CC flags respectively which are loaded from the CAW.

The on state of the CC flag prevents the current start I/O operation from being terminated at its normal end and, instead, causes a new CCW to be fetched in a similar manner to the fetching of the CCW during the initiation of start I/O.

For further information on chaining operations refer to IBM System/360 Principles of Operation, Form A22-6821.

Data Chaining

- Data chaining is initiated when the count register goes to zero during handling of 'service in'.
- When the CCW request latch is set, the I/O cycle control latch and I/O cycle latch are turned on.
- The chaining cycle latch is turned on by the I/O cycle control latch and the CCW request latch.
- The chaining counter is advanced by one during each chaining cycle.
- On the first chaining cycle, bits 0 to 15 of CCW 1 are gated into B0 and B1 data registers.
- On the second chaining cycle, the new address is loaded into the DAR.
- On the third chaining cycle the new count is loaded.
- On the last chaining cycle the new flags are loaded.

The purpose of data chaining is either to write data from different areas in main storage into the same record at the device, or to split data from one record into different areas in the main storage for a read operation. Execution of the operation at the I/O device is therefore not affected.

Data chaining is initiated when the handling of a 'service in' reduces the count register to zero and the CD flag is on. These conditions set the CCW request latch, provided that no errors are detected.

The CCW fetching now continues as soon as all I/O data cycles (if any) are taken. Another subchannel can also, at this time, be in a chaining process but only one chaining operation can occur at one time. Refer to the description of channel priority in the "Common Channel" section of this manual.

For the following description of the chaining cycles, it is assumed that the CCW request latch has turned on the I/O cycle control latch and I/O cycle latch.

Chaining Cycle 1

At RC2, the chaining cycle latch for the selected subchannel is set and the content of the COAR is loaded into the SAR. The data registers are reset and the chaining counter is advanced to one.

CCW 1 is read out to the SDR and halfword 0 is gated into the subchannel. Bits 0 to 15 of CCW 1 are then loaded into the B0 and B1 data registers.

Chaining Cycle 2

At the beginning of this cycle, the address in the COAR is again used to address main storage and CCW 1 is again read out to the SDR. The DAR, count register and B2 buffer register are reset at RC4 and the chaining counter is advanced by one to gate halfword 1 to the subchannel. Bits 16 to 31 of CCW 1 are then gated directly into bits 16 to 31 of the DAR. At the same time, bits 8 to 14 of the B1 data register are gated through the compress and into bit 14 of the DAR. Bit 15 of the B1 data register is gated directly into bit 15 of the DAR.

The B0 data register is not decoded into the Op register during a data-chaining operation, because the command from the last CCW is still used.

Chaining Cycle 3

In order to fetch CCW 2, the address in the COAR must be greater by four than for the last cycle; this is done by forcing a one from the chaining counter bit 0 to the SAR bit 29. Bits 14 to 28 are gated directly to the SAR.

CCW 2 is then read out to the SDR, and the advanced chaining counter gates halfword 1 to the subchannel. This loads bits 16 to 31 of CCW 2 into the count register.

Chaining Cycle 4

At the beginning of this cycle, the same CCW 2 is again read out to the SDR. The chaining counter is advanced by one on this cycle and both bits turn off. Halfword 0 is gated to the subchannel and the flags are loaded into the B0 data register and, later, into the flag bits.

The value held in the COAR is increased by eight to define the address of the next CCW. At WC1, the CCW request latch is turned off and fetching of the CCW is complete.

If this data chaining is during a read operation, the subchannel is now ready for a request in. For a write operation, the subchannel sends a request for data in order to have a byte of data ready in the B2 buffer register as described in "Start I/O Operation" under the heading "Data-Fetch for Write."

If 'request in' appears and the chaining routine is still in progress, the device is held on the interface until the fetching of the CCW is complete.

Command Chaining

Command chaining consists of three phases:

Analysis of incoming status byte
Fetching the CCW
Reselecting the device.

During command chaining, the new CCW fetched by the subchannel specifies a new I/O operation on the already-selected device. A new command must therefore be sent to the device, and the Op register must be reloaded.

An example of a command chaining performed in a tape drive is:

CCW 1 Write Tape	CCW 2 Backspace Tape	CCW 3 Read Tape

Analysis of the Incoming Status Byte

- Command chaining is indicated by 'channel end', 'device end' or both.
- Only 'device end' starts the fetching of the new CCW.

When a 'channel end' alone is presented at the end of an operation, the CC flag is tested. If the flag is on and no subchannel errors or halt conditions are detected, command chaining is indicated to the device by 'suppress out' being raised, followed 250 nanoseconds later by service out. After this, the subchannel waits for device end.

When 'device end' is presented for the subchannel, the CC flag is again tested. If no subchannel errors or halt conditions are detected, 'suppress out' followed by 'service out' is sent. The status modifier bit is now analyzed. If it is on, the value in the COAR is increased by eight which results in skipping one CCW. If the status modifier bit is off, the next sequential CCW is fetched. The CCW request latch is then set and fetching of the CCW starts.

Fetching the CCW

For CCW fetching, refer to the cycles described previously for the data-chaining operation. During chaining cycle 2, however, the Op register is decoded from the B0 data register for a command-chaining operation. The content of the B0 register is also gated to the B2 buffer register at the same time.

Reselecting the Device

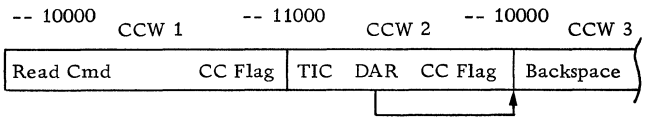
For this operation, refer to the description contained in "Start I/O Operation" under the heading "Initial Selection," together with the following details.

If the interface is free when the IF request latch is set, the 'IF requested and free' latch is set and the reselection of the device is ready to start (Figure 3B-65). Bits 3 to 7 in the DEVAR and bits 1 and 2 in the subchannel select register are now gated to bus out and the 'address out' tag is then raised.

Transfer-In-Channel

- TIC command occurs in both data and command chaining.
- The data address in the TIC command defines the next CCW to be fetched.
- The first CCW designated by the CAW must not specify a TIC command.
- A TIC-CCW must not be followed by another TIC-CCW.

A TIC command provides chaining between CCW's which are not in adjacent double-word locations in an ascending order of addresses, as for example:



When a TIC command is detected, the DAR defines the next CCW to be fetched. If the first CCW designated by the CAW is a TIC, or if a TIC-CCW follows another TIC-CCW, the 'program check' latch is set and the operation is terminated.

Recognition of TIC

- TIC command may be detected during the first CCW-fetch cycle for a chaining operation.
- TIC latch 1 is set only if no error is detected.

During the first CCW-fetch cycle for either data or command chaining, the command is inspected for a TIC. If a TIC is detected, the following conditions are tested:

The data address is tested for validity (that is, not outside storage).

The SDR parity check is tested to see if any byte in the SDR had incorrect parity during the last cycle.

TIC latch 2 is tested to determine if this is a second TIC.

Any of these conditions sets the 'program check' latch and prevents TIC latch 1 from being set. The operation then continues with the fetching of the TIC-CCW as for a normal CCW-fetch. After the completion of the CCW-fetch, however, the operation is terminated by setting the interrupt request latch.

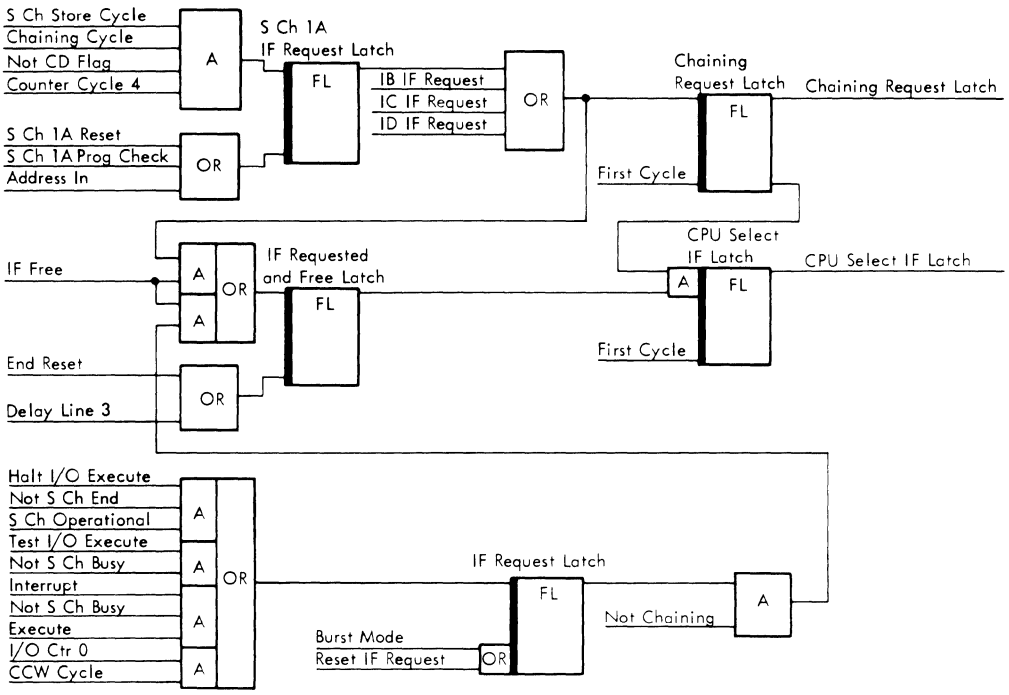


Figure 3B-65. IF Request Latches

If, however, no error conditions are detected, TIC latch 1 is set and the branch to the new CCW is started.

Branch to New CCW

- Halfword 1 of CCW 1 is read out to the SDR.
- DAR, COAR, count and flags registers are reset.
- Halfword 1 (data address) is gated to the COAR.

When TIC latch 1 is set, a new I/O cycle is requested in order to read out CCW 1 to the SDR again. The chaining counter is advanced to 0 (because of the condition of TIC latch 1), and again TIC latch 2 is set and TIC latch 1 is reset. Normally, chaining counter 0 is now active in order to reset the DAR, COAR and the count and flags registers. Chaining counter 0 is suppressed, however, and chaining counter 2 is forced instead by TIC latch 2. Note that the DAR and the count register are reset during counter cycle 2 for any chaining operations and the same circuit can now be used for a TIC.

Bits 29 to 31 in the SDR are also checked for zero, because a CCW is always located within a double-word boundary.

Halfword 1 of the SDR is then loaded into the COAR and a new I/O cycle is requested. The new CCW is fetched in the same manner as for a normal chaining operation. If this CCW also specifies a TIC command, the 'program check' latch is set and no TIC is performed.

TEST I/O OPERATION (HSMPX CHANNEL)

The purpose of the test I/O instruction is to test the state of the addressed channel, subchannel, control unit and device. The state is indicated by setting the condition code in the PSW and, in some cases, by storing the CSW.

1. A condition code of 00 is set if the addressed channel, subchannel, control unit and device is available.
2. A condition code of 01 is set if the CSW is stored.
3. A condition code of 10 is set if:
   The burst mode latch is on.
   The subchannel is busy, except when the subchannel has an interrupt pending for the addressed device.
4. A condition code of 11 is set if:
   The channel and subchannel are not operational.
   'Select in' is gated during initial selection.
5. If an interrupt is held pending in the addressed subchannel or device, the CSW is stored and indicates when the instruction is executed.

The test I/O instruction is in the SI format, and the CPU must be in the supervisor state when the instruction is executed.

I-Fetch

The I-fetch operation is described in detail in Principles of Operation - Processing Unit, Form Y33-0002.

Test I/O Execution

- CPU sequence latches turn on the compute clock.
- The channel is decoded at CC6.
- The I/O counter is advanced by I/O clock pulses.
- The I/O counter turns on the execute latch.
- The subchannels are tested.
- The I/O counter turns on the execute control latch.
- The control unit and device are tested.
- Any condition code set at any time terminates the test I/O operation.

Figure 3B-66 shows a simplified flow chart of the test I/O operation. After the I-fetch operation is completed, the CPU sequence control latch and CPU sequence latch are turned on and the compute clock is started.

At CC6, the channel is decoded and the I/O counter is advanced by I/O clock pulses.

If an invalid address is detected at this time, a condition code of 11 is set in the PSW and the operation is ended.

The execute latch is turned on at I/O counter 2 time, the burst mode latch is tested and a condition code of 10 is set if the latch is on. At I/O counter 3 time, the 'execute control' latch is set and the following conditions are tested:

1. If the selected subchannel is not operational at this time, a condition code of 11 is set and the operation is terminated.
2. If the selected subchannel is busy and an interrupt is pending in the subchannel, the address in the DEVAR is compared with the address in the B register (bits 27 to 31).

   An equal comparison indicates that an interrupt is pending in the subchannel for the addressed device. The CSW is stored and the interrupt is cleared. At the end of the last CCW cycle, a condition code of 01 is set in the PSW and the subchannel is reset.

   If the comparison is not equal, however, a condition code of 10 is set and the operation is terminated.
3. If the selected subchannel is busy and no interrupt is pending, a condition code of 10 is set and the operation is ended.

When the selected subchannel is not busy, the 'CPU request IF' latch is set at I/O counter 3 time. At the same time, the execute control latch is turned on.

'Interface free' sets the 'IF requested and free' latch and the device is now selected as described under the heading "Initial Selection" in "Start I/O Operation."

The response to 'address out' may be one of the following conditions:

1. Select in, which indicates that the addressed control unit is not available; a condition code of 11 is set in the PSW and the operation is ended.
2. Status in, which indicates either that the control unit is working or that a device other than the currently addressed one has a status stacked; the status is gated into the B2 buffer register and stored in the CSW.

   At WC1 during the CSW cycle, a condition code of 01 is set and the 'last I/O execute' latch is set. The operation is then ended.
3. Address in, which indicates either that the addressed device has an interrupt stacked or that the status of the addressed device is zero.

   The control unit generates an address to bus in and 'address in' starts the IF delay-line drive. At IF delay time, the address on bus out is compared with the address on bus in and the IF control check is set if the addresses do not match.

   Command out is sent in response to address in. Command out is detected by the control unit and 'status in' is then raised. The status byte is sent to bus in and to the B2 buffer register.

   The status byte is then analyzed for zero status or non-zero status. For a zero status, a condition code of 00 is set and the test I/O operation is ended; for a non-zero status, however, the status portion of the CSW is stored (refer to "Storing the CSW").

   At WC1 during the last CSW cycle, a condition code of 01 is set and the operation is terminated.

For condition code setting operation, refer to "Channel Condition Code Latches" in the "Common Channel" section of this manual.

HALT I/O OPERATION (HSMPX CHANNEL)

The purpose of the halt I/O instruction is to halt the current operation on the addressed subchannel and

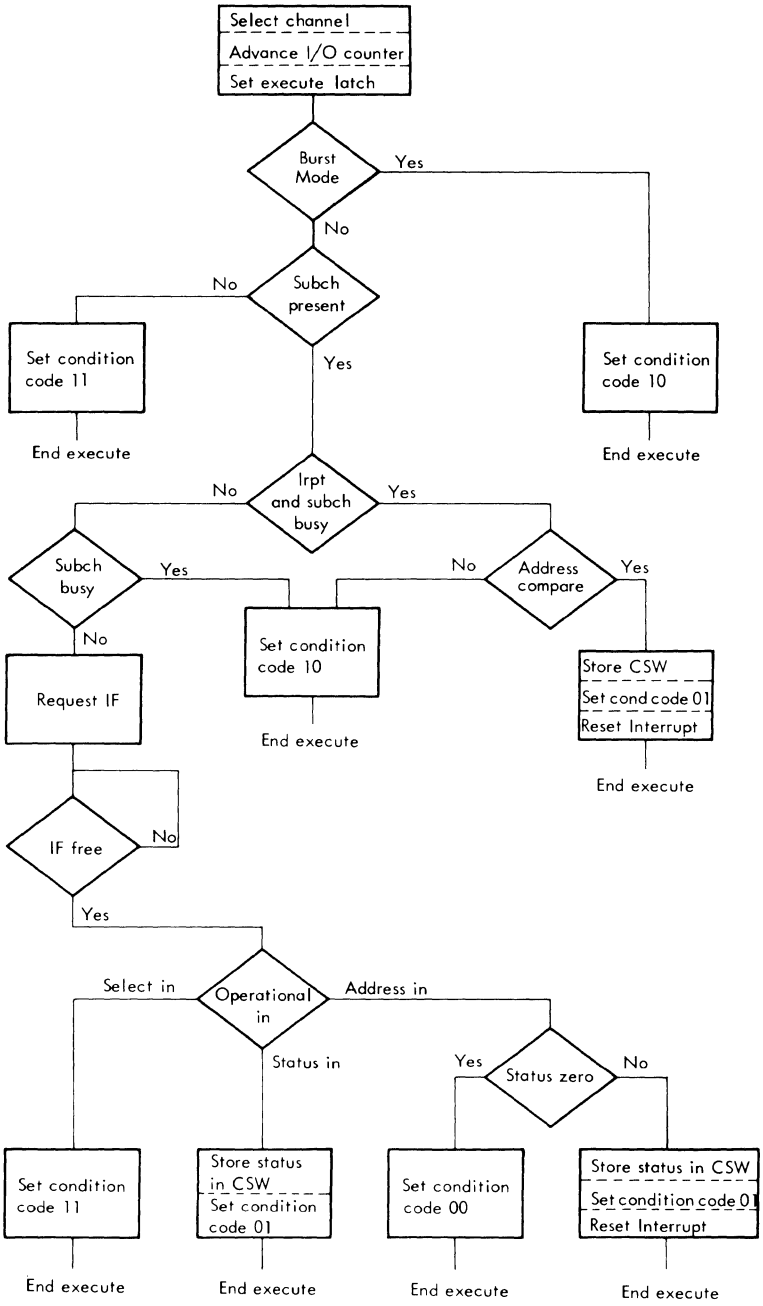


Figure 3B-66. Test I/O (HSMPX Channel)

device and to disconnect any device working in burst mode. In addition, the state of the subchannel, control unit and device is indicated by setting the condition code and, in some cases, by storing the CSW.

1. A condition code of 00 is set if the selected subchannel is busy and has an interrupt pending.
2. A condition code of 01 is set if the CSW is stored.
3. A condition code of 10 is set if a device working in burst mode was disconnected.
4. A condition code of 11 is set if:
   The channel or subchannel is not operational.
   'Select in' is gated during initial selection.

The CPU must be in the supervisor state when the instruction is executed.

I-Fetch

For details of the I-fetch operation, refer to the I-fetch information for SI format contained in Principles of Operation - Processing Unit, Form Y33-0002.

Halt I/O Execution

- The channel and subchannel are decoded at CC6.
- The I/O counter is advanced by I/O clock pulses.
- The execute latch is turned on by the I/O counter.
- The channel and subchannel are tested.
- The control unit and device are tested and halted.
- Any condition code set at any time terminates the halt I/O operation.

Figure 3B-67 shows a flow chart of the halt I/O execution. After the I-fetch operation is completed, the 'CPU sequence control 1' latch and CPU sequence latch are turned on and the compute clock is started. The channel and subchannel are decoded at CC6 and the I/O counter starts advancing by I/O clock pulses. If an invalid address is detected at this point, a condition code of 11 is set and the operation is ended.

At I/O counter 2 time, the execute latch is set and the burst mode latch is tested. If the burst mode latch is on, a condition code of 10 is set in the PSW and the operation is terminated. The execute control latch is set at I/O counter 3 time.

The following conditions are now tested:

1. Addressed subchannel operational. If the addressed subchannel is not present at this time, a condition code of 11 is set and the halt I/O operation is ended.

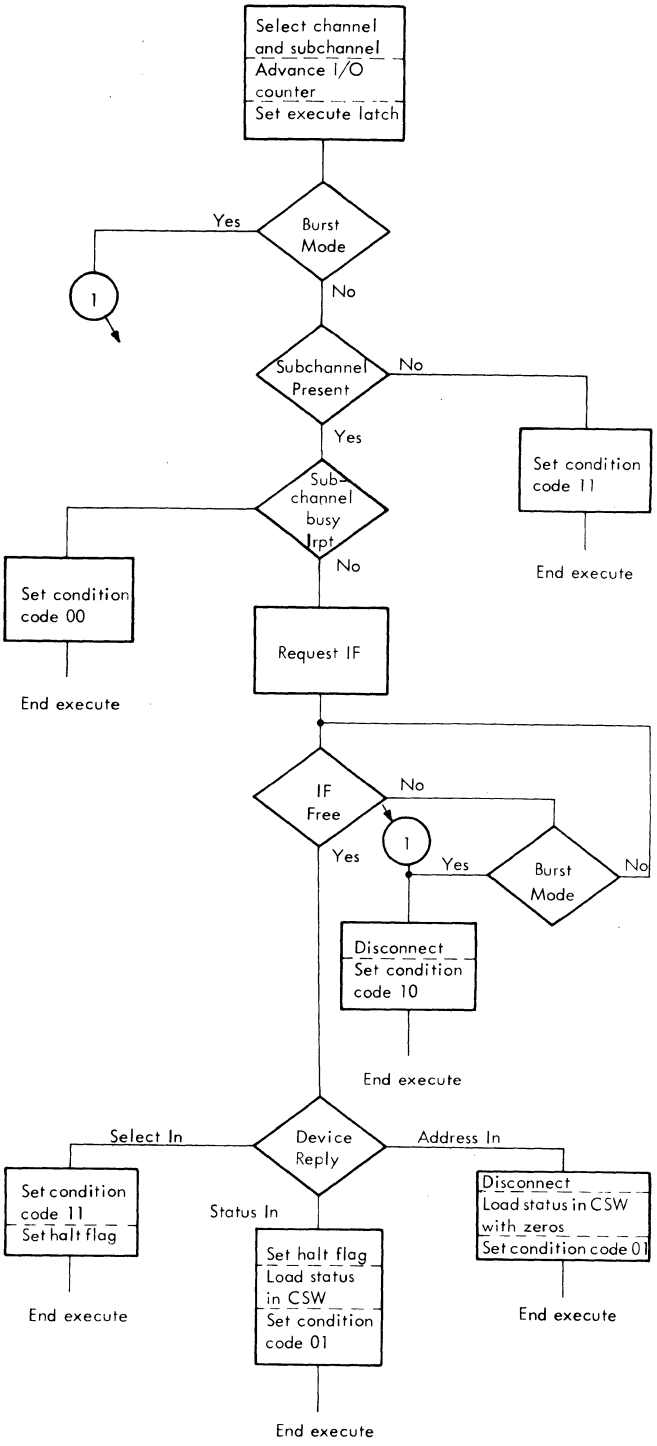


Figure 3B-67. Halt I/O

2. End status in the subchannel. If the subchannel is busy and an interrupt is pending, a condition code of 00 is set in the PSW and operation is ended. This means that a channel-end interrupt is waiting to be accepted by the CPU. However, if the subchannel has no channel-end interrupt stored, the 'CPU request IF' latch is set.

If the interface is working in byte mode, a wait condition results until the interface is free. If the interface is in burst mode, however, a condition code of 10 is set and the device which is working is disconnected from the interface (refer to "Interface Disconnect").

'Interface free' sets the 'IF requested and free' latch and the device is selected as described under the heading "Initial Selection" in "Start I/O Operation."

The response to 'address out' may be one of the following:

1. Select in. A condition code of 11 is set to indicate that the addressed unit is not available, and the operation is ended.
2. Status in, which is raised if the addressed control unit is busy. A status byte is then gated into the B2 buffer register and the status byte is stored into the CSW. A condition code of 01 is set in the PSW and the operation is ended.
3. Address in. The addressed control unit and device are disconnected from the interface. Zeros are loaded into the status part of the CSW and a condition code of 01 is set. The halt I/O operation is then ended.

Interface Disconnect

The IF disconnect is used during a halt I/O operation in order to disconnect:

A device that is working in burst mode.

The selected device during initial selection when 'address in' is presented.

The interface disconnect latches are shown in Figure 3B-68. The IF disconnect is indicated to the control unit and device by raising 'address out' without 'select out'; 'address out' must be up for at least 400 nanoseconds.

If a device is working in burst mode, the IF disconnect is initiated only after the end of tag delay 1 if 'service in' or 'status in' is up.

During an initial selection for a halt I/O, the IF disconnect is initiated when 'address in' is raised.

When 'address out' is up for more than 400 nanoseconds and 'select out' is still down, the control unit drops 'operational in'; this resets the halt I/O latch and 'address out' latch. When the 'first cycle' and 'CPU select IF' latches are reset, the 'halt I/O gate' latch is turned off the disconnect sequence is completed.

TEST CHANNEL (HSMPX CHANNEL)

The purpose of the test channel instruction is to test the state of the addressed channel. The state of the channel is indicated by setting a condition code in the PSW when the execution is ended.

1. A condition code of 00 indicates that the addressed channel is available.
2. A condition code of 01 indicates that an interrupt is held pending in the channel.
3. A condition code of 10 indicates that the channel is busy (burst mode).
4. A condition code of 11 indicates that the channel is not operational.

The test channel instruction is in the SI format, and the CPU must be in the supervisor state when the instruction is executed.

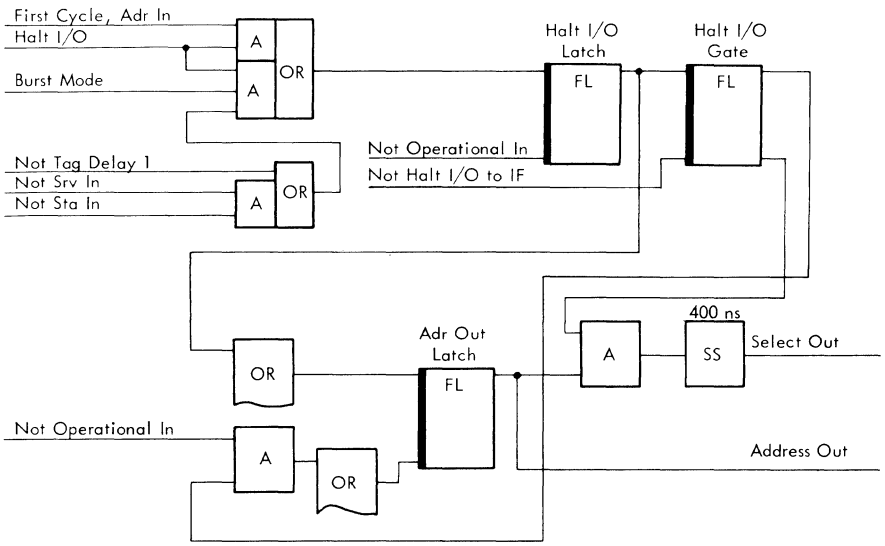


Figure 3B-68. IF Disconnect Latches

I-Fetch

The I-fetch operation is described in detail in Principles of Operation - Processing Unit, Form Y33-0002.

Test-Channel Execution

- CPU sequence latches turn on the compute clock.
- The channel is decoded at CC6.
- If an invalid address is detected, a condition code of 11 is set.
- The I/O counter is advanced by I/O clock pulses.
- The execute latch is set at I/O counter 2 time.
- A condition code of 10 is set if the burst mode latch is on.
- The execute control latch is set at I/O counter 3 time.
- A condition code of 01 is set if any subchannel has its interrupt request latch on.
- A condition code of 00 is set if the channel is available.

Figure 3B-69 shows a simplified flow chart of a test-channel operation. After the I-fetch operation is completed, the 'CPU sequence control 1' latch and CPU sequence latch are set and the compute clock is started.

At CC6, the channel is decoded. An invalid address detected at this point terminates the operation by setting a condition code of 11 in the current PSW. If the address is valid, however, the I/O counter starts to advance by I/O clock pulses.

At I/O counter 2 time, the execute latch is set and the burst mode latch is tested. If it is on, the burst mode test latch is set; a condition code of 10 is set in the PSW. The I/O counter is inhibited from being advanced when the burst mode latch is on.

If the burst mode latch is off, however, the execute control latch is set and the subchannel interrupt request latches are tested. If any of these is on, a condition code of 01 is set in the PSW and the operation is terminated.

If none of the subchannel interrupt request latches is on, a condition code of 00 is set in the PSW and the operation is terminated. For setting of the condition code, refer to "Channel Condition Code Latches" in the "Common Channel" section of this manual.

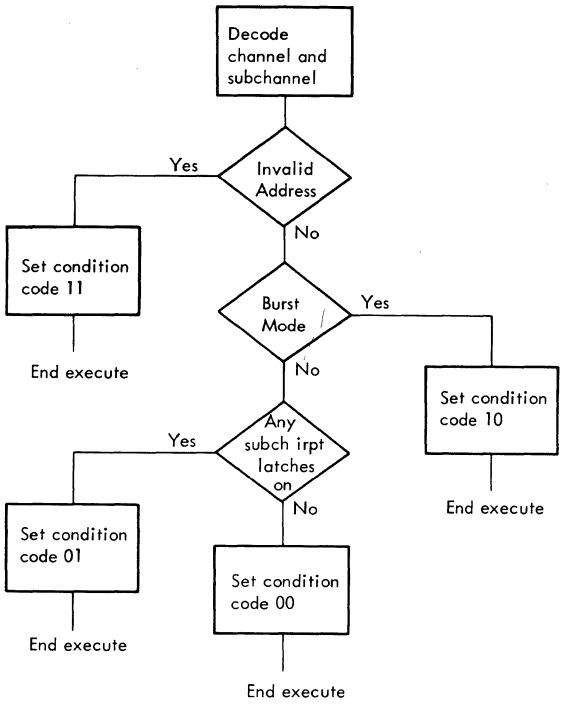


Figure 3B-69. Test Channel

STORING THE CSW

The purpose of storing the CSW is to store information from the subchannel, control unit and device into a fixed location in main storage so that this information can be analyzed later by the program. The information stored depends upon the type of current operation and the conditions of the subchannel and device.

The CSW can be stored during the execution of start I/O, test I/O and halt I/O instructions, and is always stored during an I/O interrupt.

The storing of the CSW is also the last part of the execution of start I/O, test I/O or halt I/O instructions and a condition code of 01 is always set at the end of the last CSW cycle.

The storing of the new CSW during an I/O interrupt, however, takes place after the first interrupt cycle. The interrupt cycles are then completed after the last CSW cycle.

CSW Cycles

- A CSW store latch is set to indicate that a CSW will be stored.
- The CSW latch turns on the C-cycle control latch.
- The C-cycle control latch turns on the CSW cycle latch.
- The I/O counter is advanced at WC2 on each C-cycle.
- Information is loaded from the subchannel and device to fixed locations 40 to 48 hex in main storage.
- One or four C-cycles are needed to store the CSW.

Figure 3B-70 tabulates CSW storing for the different operations and conditions and shows the CSW format. The different operations and conditions are:

1. Storing the CSW during an interrupt and a test I/O operation when the subchannel is busy.
2. Storing the CSW during an interrupt and a test I/O operation when the subchannel is not busy.
3. Storing the CSW during a program-controlled interrupt.
4. Storing the CSW during a start I/O operation and a halt I/O operation.

Interrupt and Test I/O (Subchannel Busy)

When the subchannel is busy, information is available in the subchannel which has to be stored in the CSW. For example, when a channel-end interrupt is accepted, the subchannel is always busy.

First CSW Cycle: At the beginning of this cycle, the fixed address is forced into the SAR (bits 25 and 29 are on). The on state of bit 29 means that the second word of the CSW is addressed and the word is read out to the SDR.

At this time, the I/O counter 0 gates the contents of the count register to bus in, and byte controls 2 and 3 load the contents into bytes 2 and 3 in the SDR.

The I/O counter is advanced to 1 at WC2.

Second CSW Cycle: The fixed address is gated to the SAR and the same word is read out to the SDR. I/O counter 1 gates the contents of the DAR and the channel status to bus in. Byte controls 0 and 1 load the contents into bytes 0 and 1 in the SDR.

The I/O counter is advanced to 2 at WC2.

Third CSW Cycle: This cycle starts by forcing the fixed address (bit 25 on) to the SAR and the first word of the CSW is read out to the SDR.

Bits 16 to 28 of the COAR are gated to bus in and byte controls 2 and 3 load the contents into bytes 2 and 3 in the SDR.

The I/O counter is advanced to 3 at WC2.

Type of Operation		Cycle 1 I/O Counter 0	Cycle 2 I/O Counter 1	Cycle 3 I/O Counter 2	Cycle 4 I/O Counter 3
Interrupt or Test I/O (Subchannel busy)	Gated	Count	Status	COAR	COAR
	Byte Control	2-3	0-1	2-3	0-1
Interrupt or Test I/O (Subchannel not busy)	Gated	—*	Status	—*	—*
	Byte Control	2-3	0-1	2-3	0-1
Program Control Interrupt	Gated	Status	—*	COAR	COAR
	Byte Control	1	2-3	2-3	0-1
Start I/O	Gated	Status	Only one Cycle		
	Byte Control	0-1			
Halt I/O (Control Unit	Gated	Status	Only one Cycle		
	Byte Control	0-1			
Halt I/O (Control Unit not busy)	Gated	—*	Only one Cycle. Status Portion loaded with Zeros		
	Byte Control	0-1			

* Zeros are loaded

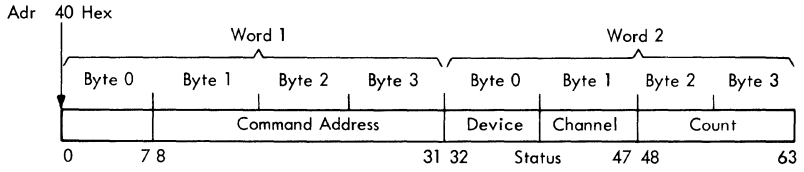


Figure 3B-70. CSW Storing Table and CSW Format

Fourth CSW Cycle: On the last CSW cycle, word 1 of the CSW is again read out to the SDR. Bits 14 and 15 of the COAR are then gated to bus in and loaded into bits 14 and 15 in the SDR by byte controls 0 and 1. Bits 0 to 13 are loaded with zeros. During this cycle, the CSW store latch is turned off. If this storing of the CSW is caused by a test I/O instruction, a condition code of 01 is set in the PSW and the test I/O execute latch is turned on.

If, however, this storing is caused by an I/O interrupt, the interrupt cycle control latch is set in order to finish the interrupt.

Interrupt and Test I/O (Subchannel Free)

When the subchannel is not busy, no information is available in the subchannel registers. However, a device status byte is stored. Subchannel free would occur, for example, on a device-end interrupt.

The storing of this CSW is similar to that for subchannel busy, except that no information is gated on the first, third and last CSW cycles. The status is gated as before on the second CSW cycle.

Program-Controlled Interrupt

Four cycles are needed to store the CSW during a program-controlled interrupt.

First CSW Cycle: The channel status is loaded into byte 1 in the SDR. There is normally no device status available because the storing of this CSW may occur when the subchannel is performing a read or write operation. This first cycle is a split cycle, since a status bit could be turned on during the time the status byte is gated to the SDR.

Second CSW Cycle: During this cycle, zeros are loaded into bits 48 to 63 in the CSW. The content of the count register is therefore not loaded.

Third and Fourth CSW Cycles: During these cycles, the contents of the COAR are loaded into the CSW. Bits 16 to 28 are loaded on the third cycle and bits 14 and 15 on the last cycle.

These two cycles are double cycles in order to prevent an I/O cycle from altering the contents of the COAR as it is stored in the CSW. (A chaining operation may start when the CSW is stored.)

Start I/O and Halt I/O

Only the status part of the CSW is stored. This takes one cycle.

Word 2 of the CSW is read out and the channel and device status are loaded into bits 32 to 47 of the CSW.

At the end of this cycle a condition code of 01 is set and the 'last execute' latch is turned on.

I/O INTERRUPT HANDLING

- The purpose of an I/O interrupt is to bring the attention of the CPU to any device.
- During the interrupt, the channel, subchannel and device address is stored in the interrupt code.
- Three types of I/O interrupt affect the subchannel:
  Channel-end interrupt.
  Device-end interrupt.
  Program-controlled interrupt.

When an I/O device or a subchannel needs the attention of the CPU, an I/O interrupt is signalled. The CPU program is then branched to another program so that the CPU is free to attend to the device or subchannel that caused the interrupt. The device and the subchannel are identified in the interrupt code after the interrupt is taken. Information is also stored in the CSW when the interrupt is complete.

Subchannel I/O Interrupts

Channel-End Interrupt

This interrupt is normally caused by a 'channel end' or a 'channel end' and 'device end' together. In both cases, the subchannel is always busy and the information to be stored in the CSW is available in the subchannel.

Device-End Interrupt

This interrupt can be caused by a device-end status byte or an attention status byte. The subchannel is always free and no information is stored in the subchannel. The status byte that caused the interrupt is stored in the control unit. The device must therefore be reselected when the interrupt is taken.

Program-Controlled Interrupt

This type of interrupt is caused by the program. The PCI flag bit is loaded from the CCW when a start I/O operation is initiated. The interrupt may then be taken as soon as allowed by the CPU (that is, when the start I/O execution is complete if the 'channel mask' bit is on). A program-controlled interrupt is taken when the subchannel is performing a read or write operation. All data from the subchannel is, therefore, not stored in the CSW. The subchannel is always busy when the interrupt is taken.

A program-controlled interrupt may also be taken together with a channel-end interrupt. In this case, the data stored from the subchannel is the same as for a channel-end interrupt.

I/O Interrupt Sequence

- An interrupt is initiated.
- The interrupt is accepted.
- First interrupt cycle is taken.
- The subchannel is tested.
- The device is reselected if subchannel is not busy.
- The CSW is stored.
- Three further interrupt cycles are taken.

Initiating an Interrupt

An interrupt is initiated by setting the respective channel request latch. The interrupt request latch may be set either from the subchannel interrupt request latches or from the subchannel PCI flag bits. See Figure 3B-71.

Accepting the I/O Interrupt

The acceptance of an I/O interrupt is described in Principles of Operation - Processing Unit, Form Y33-0002.

Selecting the Subchannel

- The subchannel is selected when the interrupt is accepted by CPU.
- More than one subchannel interrupt request latch may be on at the same time.
- The subchannel with the highest priority is selected first.

When the CPU has accepted an I/O interrupt for one channel, an interrupt accept signal is sent to the appropriate channel. The subchannel with its interrupt request latch on is then selected. However, more than one interrupt request latch may be on at the same time and it is, therefore, necessary to select first the subchannel with the highest priority. Subchannels A, B and C each have one priority latch (Figure 3B-71), which can be set from either the interrupt request latch or the respective PCI flag. As long as the interrupt accept signal is not gated to

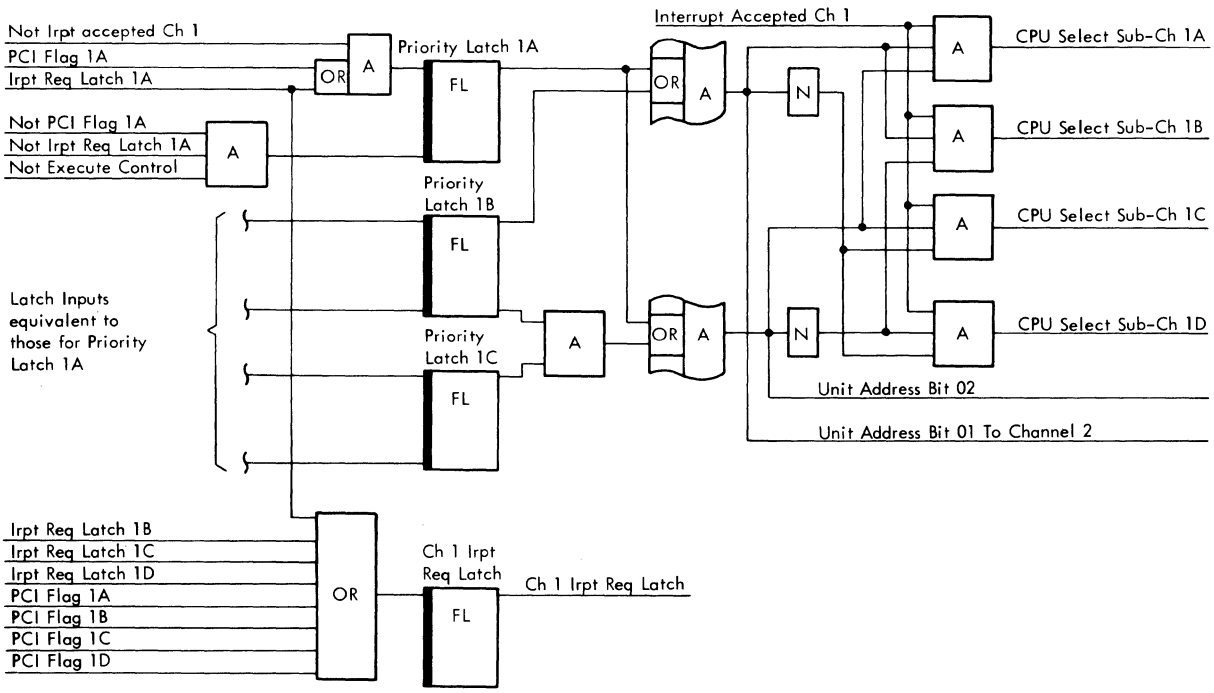


Figure 3B-71. Subchannel Priority

the channel, the priority latches may be set at any time. Thus the state of the priority latches cannot be altered during the time the interrupt accept signal is gated.

The output of the priority latches is gated into a priority circuit and the output of this circuit is used to select the correct channel and subchannel.

If none of the priority latches is on, but an interrupt accept signal is gated, subchannel D is selected. When the subchannel is selected, the busy latch is tested.

Subchannel Interrogation

- The I/O counter is advanced by I/O clock pulses.
- The execute latch is set at I/O counter 2 time.
- The execute control latch is set at I/O counter 3 time.
- The CSW store latch is set if the subchannel is busy.
- The IF request latch is set if the subchannel is not busy.

The interrupt accept signal starts the I/O counter advancing by I/O clock pulses to test the selected subchannel. At I/O counter 2 time, the execute latch is turned on. The execute control latch is set one I/O clock pulse later and the subchannel is tested. If the subchannel is busy, the 'CSW store' latch is set and the loading of the CSW starts immediately. If the subchannel is not busy, however, it means that the status byte to be stored in the CSW is not available in the subchannel. The status byte must therefore be fetched from the control unit.

Reselecting the Device

- The device is reselected in order to fetch a status byte.
- Bits 1 and 2 from the interrupt-priority circuit and bits 3 to 7 in the DEVAR are used as device address.

This operation is similar to an initial-selection sequence for a start I/O operation, except that bits 3 to 7 of the address that is used to select the control unit and device are taken from the DEVAR. When the status byte is gated to bus in and loaded into the B2 buffer register, the STAN latch is set. The analysis of the status byte results in the CSW store latch being set and 'service out' being raised in order to clear the status byte in the control unit. The storing of the CSW then takes place. If the interface is working in burst mode when the interface is requested, or if 'select in' is gated during the initial-selection sequence, the cancel interrupt latch is set. 'End execute' is then forced and the selected subchannel is reset (including the interrupt-request and suppress-out latches). The channel interrupt latches are also reset. When the interface is next free, the control unit presents the same status byte, and an I/O interrupt is again requested.

CSW Storing

For details of this operation, refer to "Storing the CSW." During the last CSW store cycle, the interrupt cycle control latch is set in order to complete the three interrupt cycles. During the old PSW 1 period, the selected subchannel is reset.

INITIAL PROGRAM LOADING (HSMPX CHANNEL)

The purpose of the IPL operation is to start the loading of a program from any device. This operation is divided into the following phases (summarized in Figure 3B-72):

System reset
Channel and subchannel decoding
CCW fetching
Initial selection
Data transfer
Command chaining
Storing of the interrupt code
PSW loading.

System Reset

System reset is described in Section 2.9.2 of "Console" in FEMM IBM System/360 Model 44, Form Y33-0007.

Channel and Subchannel Decoding

At the end of the system reset, the channel and subchannel are decoded from the load unit address switches on the console.

The channel is selected and the appropriate channel decode latch is turned on to select the subchannel. At the same time, the I/O counter is advanced by I/O clock pulses and, at I/O counter 2 time, the execute latch, IPL latch 1 and IPL latch 2 are turned on. At I/O counter 3 time, the execute control latch is turned on in order to execute the IPL operation. The execute latch is reset at I/O counter 0 time, which prevents the I/O counter from being advanced by I/O clock pulses. The IPL 2 latch and the execute control latch turn on the CCW latch at I/O counter 0 time if the subchannel is operational.

	Phases	Comments
1	System Reset	System reset; see Section 2.9.2 of "Chapter 2 Console" FEMM IBM System/360 Model 44, Form Y33-0007.
2	Channel and Subchannel Select	Channel and subchannel are decoded from bits 22 and 23 set up by LUA switches.
3	CCW Forcing	Four I/O cycles. Count register set to 24. Op register set to 'read forward'. SLI and CC flags set.
4	Initial Selection	Device set up by LUA switches is selected.
5	Read Operation	24 bytes are transferred into locations 00 to 24 in main storage.
6	Command Chaining	Next CCW is loaded into subchannel. Four I/O cycles.
7	Initial Selection	Device reselected using address in DEVAR.
8	Read Operation	Transfer data to main storage as indicated by new CCW.
9	Channel and Device End	CC flag on; enter chaining routine (see item 6). 'Device end' and no chaining flag set 'interrupt' latch.
10	Store Interrupt Code	One C-cycle taken to store interrupt code into location zero in main storage.
11	Load PSW	PSW loaded from location zero in main storage, and CPU continues with this information.

Figure 3B-72. Sequence of IPL Operation

CCW-Fetch

The fetching of the CCW during an IPL operation is, in many ways, similar to the CCW-fetching during a start I/O operation, and sequences of the I/O cycles are the same. However, control information is not fetched from main storage but is forced from the halfword-select (Figure 3B-43). The output of the SDR must therefore be suppressed.

CCW Cycle 1

At the beginning of this cycle, all the registers in the subchannel are reset. The contents of the COAR (now zero) are gated to the SAR and the first word is read out to the SDR. The I/O counter is advanced to 1 and bit 6 in the B0 data register is forced to a one. All the other bits in the B0 and B1 data registers are zero.

CCW Cycle 2

The same word is again read out and the I/O counter is advanced. During this cycle, the halfword-select gates zeros to the DAR.

The B0 data register is also decoded and a read-forward operation is set into the Op register.

CCW Cycle 3

The second word is now read out to the SDR, and the I/O counter is again advanced. The halfword-select gates bits 27 and 28 to the count register, thus equalling a byte count of 24. The contents of the B0 data register are also gated into the B2 buffer register.

CCW Cycle 4

During the last cycle, the same second word is again read out to the SDR and the I/O counter is now advanced to 0. Bits 1 and 2 are now forced from the halfword-select and loaded first into the B0 data register, then to the CC and SLI flags. The value in the COAR is increased by eight and the CCW latch and CCW cycle latch are turned off.

The subchannel is, at this time, set up to read 24 bytes into location zero in main storage.

Initial Selection

The selection of the device during an IPL operation is similar to selection during a start I/O operation with the following two exceptions:

The address to bus out is gated from the console load unit address switches.

Any condition code set does not set the 'last I/O execute' latch.

The IPL latch 1 is turned off on completion of initial selection.

Data Transfer

When the device is selected, 24 bytes are transferred through the subchannel and loaded into main storage in locations 00 to 18 hex.

The device-end condition causes the command-chaining routine to be entered because the CC flag is on. The status analysis and the read operation are described in "Start I/O Operation" under the headings "Status Analysis during Initial Selection" and "Data Service."

Command Chaining

The fetching of the new CCW during an IPL operation is similar to that described for a start I/O operation.

The new CCW is fetched from locations 08 to 10 hex in main storage. At the end of CCW-fetch, the same device is again selected and new data is read into main storage. This sequence now continues until a channel-end condition is detected and the CC flag is off. The channel-end condition then sets the interrupt request latch which turns on the CSW store latch to store the interrupt code.

Storing the Interrupt Code

The on state of the CSW store latch turns on the C-cycle control latch, which then sets the CSW cycle latch. Address zero is forced to the SAR and the word is read out. During this cycle the device, subchannel and channel address is gated to the SDR. Byte controls 2 and 3 load the address into bytes 2 and 3 in the SDR.

The subchannel is reset at WC1, and the CSW store latch is turned off. At the end of this cycle, the load-PSW routine is entered by forcing the line 'load PSW op'.

Loading the PSW

The load-PSW operation is described in Principles of Operation - Processing Unit, Form Y33-0002.

During the first EA-cycle, the new PSW 1 in location 00 of main storage is read out to the SDR, then gated through the B register and into PSW 1. The execute control latch and the IPL latch 2 are turned off, resulting in the setting of the 'last I/O execute' latch.

On the second EA-cycle, location 04 in main storage is read out to the SDR and the contents are gated to the PSW 2. The CPU now proceeds with the new information in the PSW.

CHECKING (HSMPX CHANNEL)

- Five check latches are provided in each subchannel.

For the HSMPX channel, parity is carried between SDR and the interface. The parity on bus in and bus out is therefore checked. The check latch that is set depends on the type of information on the bus at the moment the check is detected.

The CAW and CCW format is always checked at the time the CAW and the CCW are fetched from main storage and loaded into the channel. A format check sets the 'program control check' latch.

The number of bytes transferred over the interface for one read operation is always checked against the byte count requested by the channel. If the number of bytes in the record is not equal to the byte count the IL indicator is set, provided that the SLI flag is off.

A machine malfunction error detected in the channel sets the IF control check. An example of a machine malfunction is the simultaneous raising of status in and service in.

Each subchannel has five check latches which together with the PCI flag, constitute the subchannel status. The latches are:

Incorrect Length (IL) latch
Channel Data Check (CDC) latch
Program (PROG) check latch
Channel Control Check (CCC) latch
Interface Control Check (ICC) latch.

Incorrect Length Latch

- Set to indicate a wrong-length record.
- The setting of the IL latch is suppressed if the SLI flag is present.
- The IL latch is not an active check latch.
- The IL check is indicated in the CSW when the I/O interrupt is taken.

The logic of this latch is shown in Figure 3B-73. The IL latch is set when the number of bytes contained in storage for an I/O operation does not equal the number of bytes requested by the I/O device. The latch is set either at the time a status byte is being presented and the count is not zero, or at the time 'command stop' is gated to the device. The IL latch is also set if the CD flag is on at the time a status byte is presented to the subchannel.

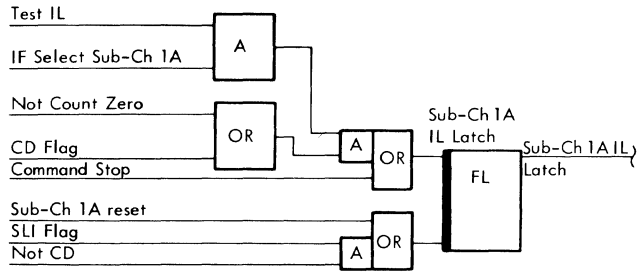


Figure 3B-73. Incorrect Length Latch

The on state of the SLI flag prevents the setting of the IL latch unless the CD flag is on.

The IL latch is not an active check latch. Therefore, no action is taken in the subchannel when the check is set. The IL check is, however, loaded into the CSW during the I/O interrupt and the program may then continue.

Channel Data Check Latch

- Set to indicate channel parity errors.
- The channel data check latch is not an active check latch.

The logic associated with the channel data check latch is shown in Figure 3B-74. The latch is set to indicate parity errors detected during channel operations. Three conditions cause this latch to be set:

Parity error detected on bus out during a write operation.

Parity error detected on bus in during a read operation.

An SDR parity check during a channel cycle.

The channel data check latch is not an active check latch. Its condition is recorded in the CSW during an I/O interrupt.

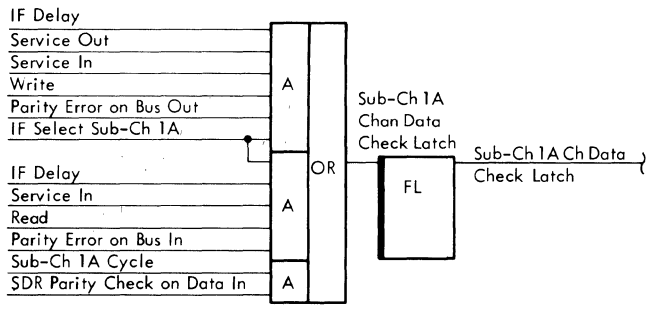


Figure 3B-74. Channel Data Check Latch

Program Check Latch

- Set to indicate a CAW or CCW format error.
- Set to indicate that an address has exceeded the capacity of main storage.
- Set in order to terminate an I/O operation.
- The program check latch is an active check latch.
- The program check is indicated in the CSW when the I/O interrupt is taken.

The logic associated with the program check latch is shown in Figure 3B-75.

CAW or CCW Format Errors

The program check latch is set during the fetching of the CAW and CCW if any of the following conditions are detected:

CCW address outside storage.
Bits 29 to 31 in the CAW are not zeros.
Bits 0 to 7 in the CAW are not zeros.
Bits 5 to 7 in the CCW 2 are not zeros.
Bits 4 to 7 in the CCW 1 are zeros except for data chaining.
A TIC in the first command.
A second TIC.
The content of the count register is zero after it has been loaded.

Invalid Address

The program check latch is also set if an address has exceeded the capacity of main storage; this condition is detected in the CPU and the signal is sent to the channel to set the program check latch (see Figure 3B-76). An invalid address may be generated when the value in either the COAR or the DAR has been increased. When an invalid address is gated to the SAR, the word defined by the address in the SAR is still read out to the channel. Therefore, all four bytes in the SDR are loaded from storage, then regenerated by inhibiting the gating of the byte control. The setting of the program check latch caused by an invalid address depends on the three following operations:

1. During a CCW-fetch for chaining. The address of the CCW has exceeded the capacity of main storage and thus the program check latch is set immediately. The operation is then terminated at the end of the CCW-fetch.
2. During a read operation. The data address is outside storage and the program check latch is there-

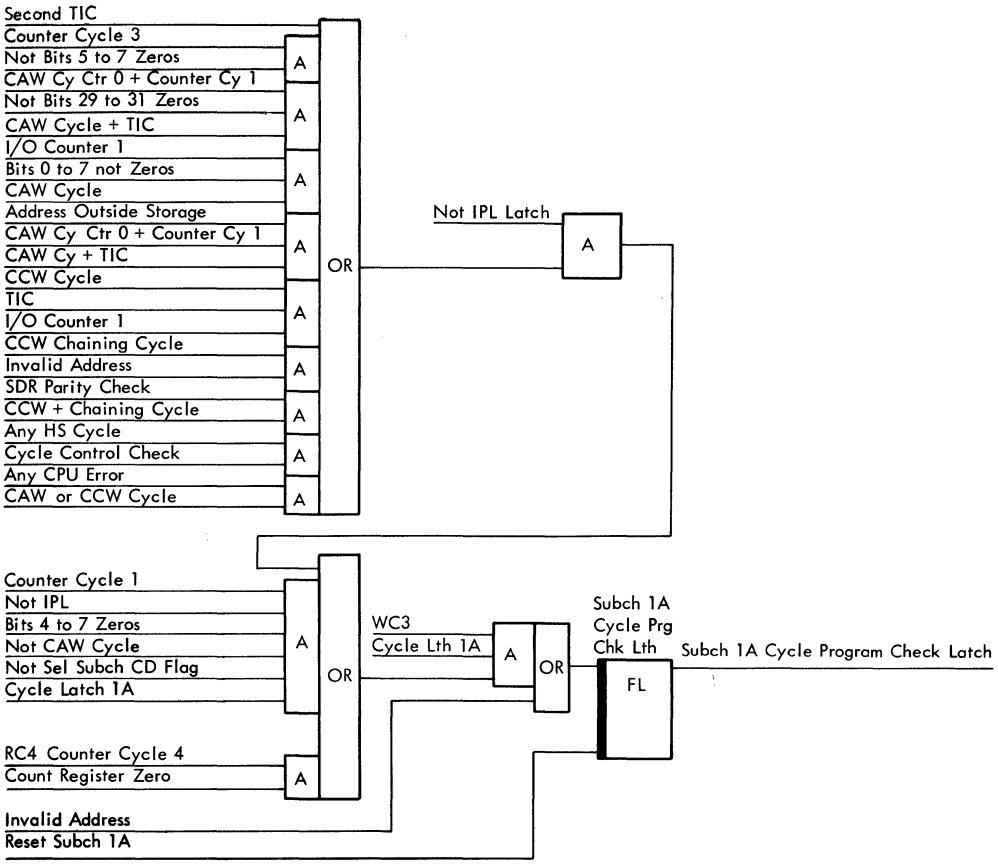


Figure 3B-75. Program Check Latch

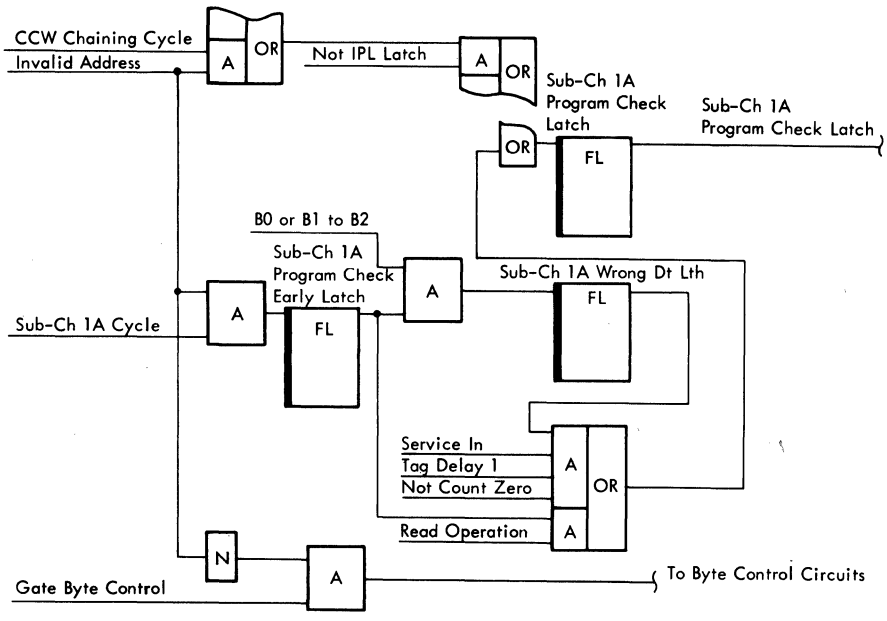


Figure 3B-76. Program Check set by Invalid Address

fore set to indicate that no bytes were loaded into main storage. When the next 'service in' appears, 'command stop' is gated to the control unit and the operation is terminated.

3. During a write operation. If an invalid address is detected during an I/O write cycle, the two bytes that have been loaded to B0 and B1 data registers are not the data requested by the channel. The program check latch must therefore be set if either of these two bytes is gated to bus out. An invalid address sets the 'program check early' latch for the selected subchannel. If either B0 or B1 data register is gated to B2 buffer register, the 'wrong data' latch is set to indicate that B2 register has the wrong data stored in it. The program check latch is set during a later 'service in' sequence, provided that the count is not zero. Count zero means that the data in B2 buffer register is not gated to bus out but a stop command is sent to the device.

Machine Malfunctions

The program check latch is further set for the following:

1. Any CPU errors detected during the CAW-fetch and CCW-fetch operations.
2. An SDR parity check detected during CCW-fetch for chaining.
3. A cycle control check detected during any subchannel cycle.

These conditions also set the channel control check latch which, in turn, sets the machine check interrupt request latch in the CPU (see "Channel Control Check Latch"). The program check latch is set in these cases to ensure that the I/O operation is terminated at the end of the CCW-fetch for either a start I/O operation or a chaining operation.

For example, assume that an SDR parity check is detected during the first CAW-fetch cycle. The program check latch and the channel control check latch are turned on, the channel control check resulting in a request for a machine check interrupt. If the machine check mask bit is on at this time, 'end execute' is forced in the CPU and the machine check interrupt is taken when the CAW cycle is finished. If the machine check mask bit is off, however, the normal sequence of the start I/O operation continues until the fetching of the CCW is completed. The program check latch is then tested, and the subchannel-status is stored into the CSW. The start I/O operation is then terminated.

Channel Control Check Latch

- Set to indicate machine malfunction.
- The channel control check latch is an active check latch.
- The channel control check latch sends a request for an external machine check interrupt to the CPU.

The logic associated with the channel control check latch is shown in Figure 3B-77. The following conditions set the latch:

1. A cycle control check during any subchannel cycle.
2. An SDR parity check during a CCW-fetch for chaining.
3. Any CPU errors detected during the CAW-fetch and CCW-fetch for a start I/O operation.
4. Incorrect parity detected when the command code is gated to bus out.

As soon as the channel control check latch is set, a signal is sent to the CPU to request a machine check interrupt.

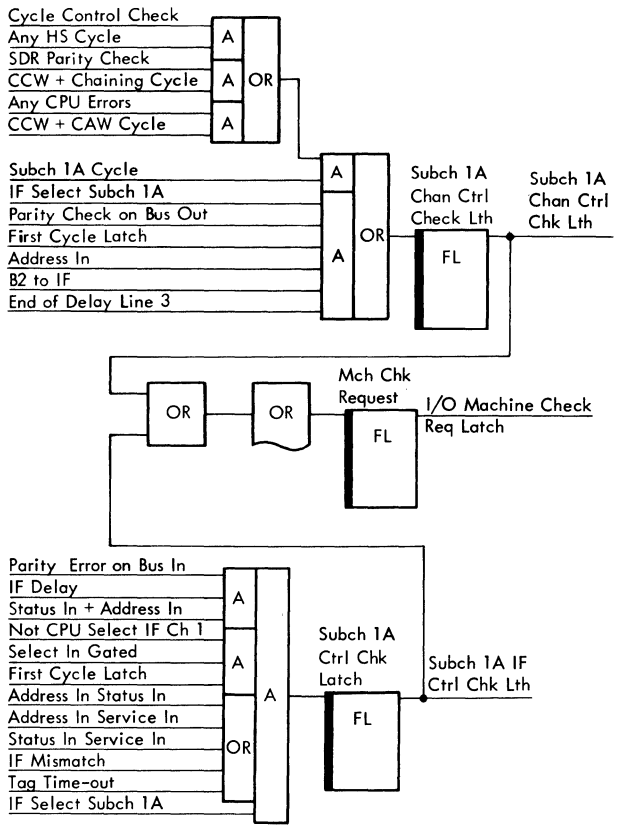


Figure 3B-77. Channel Control Check and IF Control Check Latches

Interface Control Check Latch

- Set to indicate machine malfunction.
- The interface control check latch is an active check latch.
- The interface control check latch sends a request for an external machine check interrupt to the CPU.

The logic of the interface control check latch is shown in Figure 3B-77. The latch is set if any malfunction errors are detected on the interface, such as:

1. Parity error on bus in for 'status in' or 'address in'.
2. 'Address in' and 'status in' active simultaneously.
3. 'Address in' and 'service in' active simultaneously.
4. 'Status in' and 'service in' active simultaneously.
5. Interface mismatch.
6. Tag time-out.
7. 'Select in' is gated during reselection of a device for command chaining.

The interface control check latch also requests the CPU for a machine check interrupt.

Result of a Machine Check Interrupt

- Any channel may set the external machine check request.
- Both channels may have their check latches on at the same time.
- The channel with its check latch on during the machine check interrupt is reset.

When a channel control check or an interface control check is detected in either channel, the 'I/O channel check request' latch is turned on in the CPU. At the time the machine check interrupt is taken, a signal is sent back to test the channel control check latches or interface control check latches of both channels. If any of these latches are on for any subchannel, a check reset latch is turned on and the timer flip-flops are reset (see Figure 3B-78). The output of the check reset latch then resets the channel, the duration of this reset varying between 20 microseconds (μs) and 40 μs, as shown. The check reset latch is always reset when flip-flop 2 goes on. The output of the check reset latch also inhibits the I/O counter advance and thus hangs up any new I/O operation initiated on the same channel until the check reset is completed.

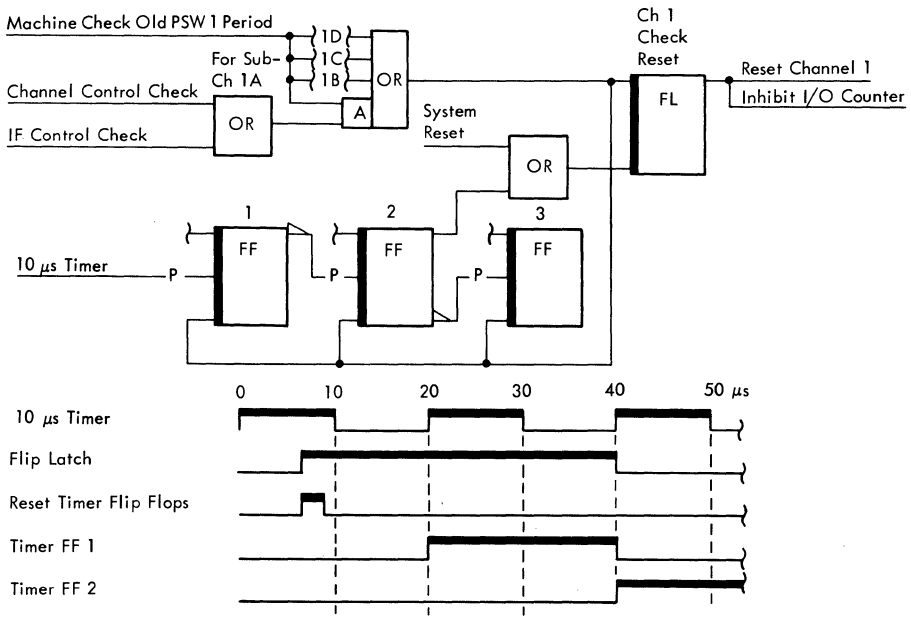


Figure 3B-78. Channel Reset by Machine Check

CUT ALONG LINE

COMMENT SHEET

System/360, Model 44 Principles of Operation -- Channels
Field Engineering Theory of Operation Y33-0003

FROM

NAME _____ OFFICE/DEPT NO. _____

CITY/STATE _____ DATE _____

To make this manual more useful to you, we want your comments: what additional information should be included in the manual; what description or figure could be clarified; what subject requires more explanation; what presentation is particularly helpful to you; and so forth.

How do you rate this manual: Excellent ____ Good ____ Fair ____ Poor ____

Suggestions from IBM Employees giving specific solutions intended for award considerations should be submitted through the IBM Suggestion Plan.

NO POSTAGE NECESSARY IF MAILED IN U.S.A.

FOLD ON TWO LINES (LOCATED ON REVERSE SIDE), STAPLE AND MAIL

fold fold

FIRST CLASS
PERMIT NO. 81
POUGHKEEPSIE, N.Y.

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

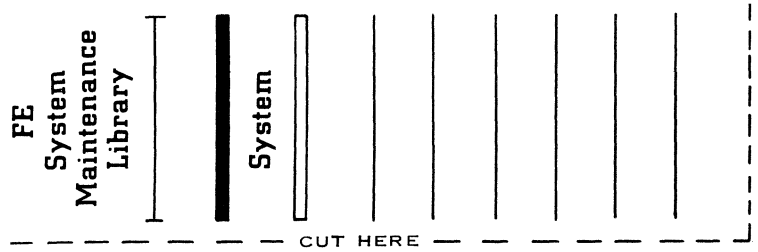
POSTAGE WILL BE PAID BY . . .

IBM CORPORATION

P.O. BOX 390
POUGHKEEPSIE, N.Y. 12602

ATTENTION: FE MANUALS, DEPT. B96

fold fold



Y33-0003-0

S/360 Model 44 Printed in U.S.A. Y33-0003-0

IBM

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