File No. S360-30
Form Y28-6616-4

Program Logic

IBM System/360 Operating System

Input/Output Supervisor

Program Number 360S-CI-505

This publication describes the operation of the I/O supervisor within the IBM System/360 Operating System control program. The I/O supervisor's components, the EXCP supervisor and the I/O interruption supervisor, are discussed in detail to show the internal structure and logic involved in the control of I/O devices and channels. In addition, error recovery procedures are described.

This program logic manual is directed to the IBM customer engineer responsible for program maintenance. It can be used to locate specific areas of the program, and it enables the reader to relate these areas to the corresponding program listings. Because program logic information is not necessary for program operation and use, distribution of this manual is restricted to persons with program-maintenance responsibilities.

Restricted Distribution

Preface

This publication discusses the operation of the I/O supervisor. It is directed primarily to IBM customer engineers and to IBM system programmers.

The publication is divided into an introduction and five sections to provide both general and detailed presentations of the following areas:

- EXCP supervisor
- I/O interruption supervisor
- Error routines
- SVC transient area routines
- Control blocks, tables, and queues

All options that may be added to the control program are included in the presentations. These are multiprogramming with a fixed number of tasks (MFT or Option 2) and multiprogramming with a variable number of tasks (MVT or Option 4). The control program options are explained in the publication IBM System/360 Operating System: Storage Estimates, Form C28-6551.

RECOMMENDED PUBLICATIONS

Information required for an understanding of this manual is contained in the following publications:

IBM System/360 Principles of Operation, Form A22-6821

For primary control program (PCP) users,

IBM System/360 Operating System:

Introduction to Control Program Logic, Program Logic Manual, Form Y28-6605

Fixed-Task Supervisor, Program Logic Manual, Form Y28-6612

For MFT users,

IBM System/360 Operating System:

Introduction to Control Program Logic, Program Logic Manual, Form Y28-6605

Fixed-Task Supervisor, Program Logic Manual, Form Y28-6612

Fifth Edition (November, 1968)

This is a major revision of, and obsoletes, Y28-6616-3, and Technical Newsletter Y28-2310.

This edition adds descriptions of the Model 65 Multiprocessing option. In addition, the descriptions of the following routines have been changed to reflect changes in the program: Tape Error routine, 2540 Error routine, Write-to-Operator routine, and the I/O Inboard and Outboard Recorders. A dictionary of acronyms and abbreviations has been added to the manual as Appendix E. The flowcharts and the index have been improved. Changes to the text and small changes to illustrations are indicated by a vertical line to the left of the change. Changes or added illustrations are denoted by the symbol • to the left of the caption.

This edition applies to release 17 of IBM System/360 Operating System and to all subsequent releases until otherwise indicated in new editions or Technical Newsletters. Changes are continually made to the specifications herein; before using this publication in connection with the operation of IBM systems, consult the latest IBM System/360 SRL Newsletter, Form N20-0360, for the editions that are applicable and current.

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

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Programming Systems Publications, Department D58, PO Box 390, Poughkeepsie, N. Y. 12602

Control Program With Option 2, Program Logic Manual, Form Y27-7128

For MVT users,

IBM System/360 Operating System:

MVT Control Program Logic Summary, Form Y28-6658

MVT Supervisor, Program Logic Manual, Form Y28-6659

In addition, the following publications may be useful for reference:

IBM System/360 Operating System:

System Control Blocks, Form C28-6628

System Programmer's Guide, Form C28-6550

Supervisor and Data Management Services, Form C28-6646

Input Output Support (Open, Close, EOV) Program Logic Manual, Form Y28-6609

Messages and Codes, Form C28-6631

IBM Printer Keyboard Model 7 with IBM 2150 Console, Form A22-6877

IBM 1403 Printer Component Description, Form A24-3073

IBM 1443 Printer, Models 1, 2, and N1 IBM 1445 Printer, Models 1 and N1, Form A24-3120

IBM 1442 N1 and N2 Component Description and Operating Procedures, Form A21-9025

IBM System/360 Component Description IBM 2250 Display Unit, Model 1, Form A27-2701

IBM System/360 Component Description IBM 2250 Display Unit, Model 2 IBM 2840 Display Control, Model 1, Form A27-2702

IBM System/360 Component Description IBM 2260 Display Station IBM 2848 Display Control, Form A27-2700

IBM 2301 Drum Storage IBM 2820 Storage Control Component Descriptions, Form A22-6895

IBM System/360 Component Descriptions
2841 Storage Control
2302 Disk Storage, Models 3 and 4
2311 Disk Storage Drive
2321 Data Cell Drive
2303 Drum Storage
Form A26-5988

IBM 2321 Data Cell Drive Original Equipment Manufacturers' Information, Form A26-3574

IBM 2400 and 2816 Model 1 Component Description, Form A22-6866

IBM 2501 Models B1 and B2 Component Description and Operating Procedures, Form A21-9026

IBM 2520 B1, B2, and B3 Component Description and Operating Procedures, Form A21-9027

IBM 2540 Component Description and Operating Procedures, Form A21-9033

IBM 2671 Paper Tape Reader, Form A24-3388

IBM 2821 Control Unit, Form A24-3312

IBM 2301 Drum Storage IBM 2820 Storage Control Component Descriptions, Form A22-6895

Contents

Illustrations

Figures

Tables

Charts

Introduction

The input/output (I/O) supervisor is the portion of the control program that issues the privileged I/O instructions and supervises the resulting I/O operations for any program that requests I/O device activity. The I/O supervisor has two purposes:

- To handle I/O requests, which are requests for the execution of channel programs.
- To handle I/O interruptions, which result from the execution of channel programs and from operator intervention.

To facilitate the handling of the I/O requests and interruptions, the I/O supervisor is divided into two major program sections:

- Execute channel program (EXCP) supervisor.
- Input/output (I/O) interruption supervisor.

These major sections reside in main storage and provide control program support for the execution of channel programs.

The EXCP supervisor handles all I/O requests issued by means of the EXCP macro instruction and SVC 15. The I/O interruption supervisor handles all I/O interruptions, provides error recovery supervision, indicates to the user when an I/O request is complete, and restarts available devices and channels.

Routines and control areas used with the I/O supervisor are:

- Error routines
- Purge routine and restore routine
- DEVTYPE routine
- IOHALT routine
- Control blocks, tables, and queues

Error routines are transient and are called by the I/O supervisor to handle device and channel errors and unusual conditions. The purge and restore routines are also transient and are used, respectively, to remove from the system I/O requests that are active or waiting to be started and to reschedule I/O requests that were previously removed. The DEVTYPE routine passes information concerning I/O device characteristics to the user. The IOHALT routine allows the graphics or telecommunications user to halt I/O activity on any device except a direct access device.

Control blocks, tables, and queues all reside in main storage. Control blocks are created either by the user at assembly time by means of control statements and macro instructions, or by the system at system generation time. Tables, and also the control fields for queues and the elements to be queued, are created at system generation time. The queues are then maintained by the I/O supervisor.

Figure 1 shows the organization of the I/O supervisor within main storage.

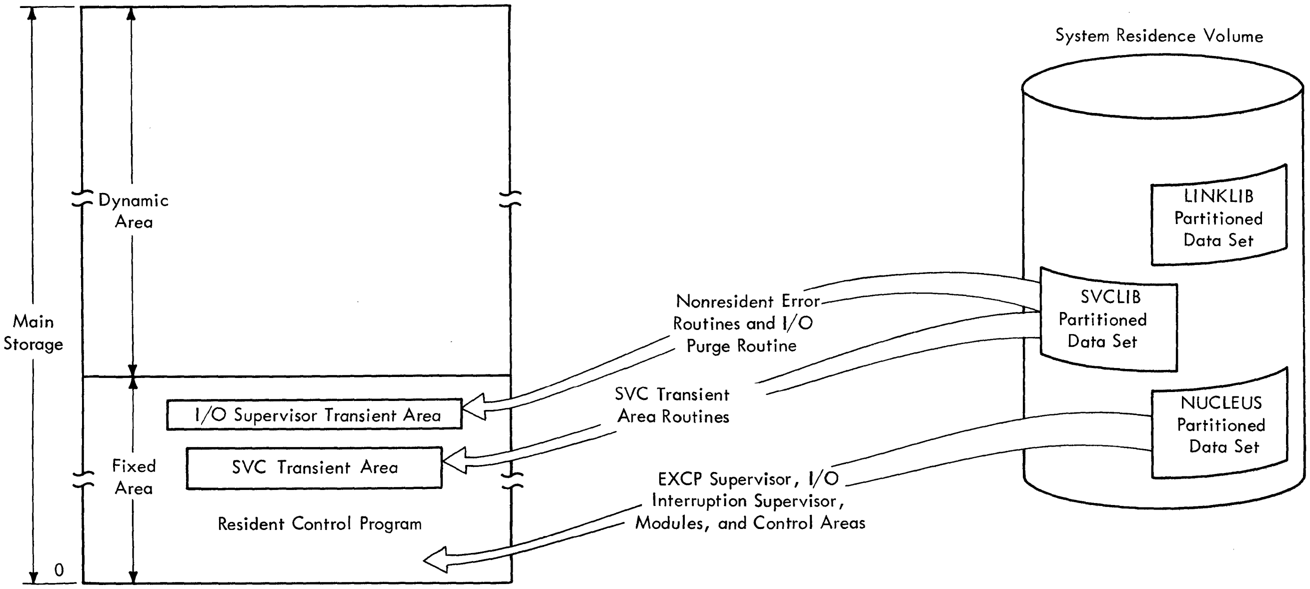

System Residence Volume
LINKLIB Partitioned Data Set
SVCLIB Partitioned Data Set
NUCLEUS Partitioned Data Set
Dynamic Area
Main Storage
Nonresident Error Routines and I/O Purge Routine
I/O Supervisor Transient Area
SVC Transient Area Routines
SVC Transient Area
Fixed Area
EXCP Supervisor, I/O Interruption Supervisor, Modules, and Control Areas
Resident Control Program
0

Figure 1. Organization of the Input/Output Supervisor Within Main Storage

The Input/Output Supervisor

The I/O supervisor starts and supervises I/O operations for programs that request I/O device activity.

Processing programs and the control program request I/O device activity via a macro instruction. The type of macro instruction that is used depends upon the use of an access method.[1]

If an access method is used, the expansion of the GET/PUT or READ/WRITE macro instruction contains a branch to access method routines. These routines construct a channel program, provide control block information, and cause control to be routed to the I/O supervisor via an EXCP macro instruction.

If an access method is not used to request the execution of channel programs, the processing program or the control program must issue the EXCP macro instruction. | Before doing so, it must provide the necessary channel program and the control block information required for the I/O supervisor.[2]

Figure 2 illustrates the relationships among a processing program or the control program, an access method, and the I/O supervisor.

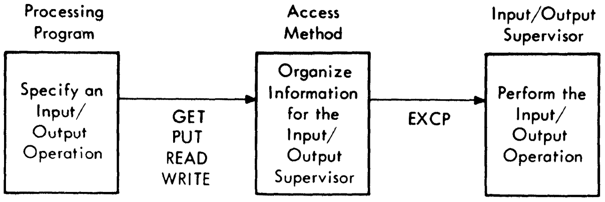

Processing Program
Specify an Input/ Output Operation
GET
PUT
READ
WRITE
Access Method
Organize Information for the Input/ Output Supervisor
EXCP
Input/Output Supervisor
Perform the Input/ Output Operation

Figure 2. Relationships Among a Processing Program or the Control Program, an Access Method, and the Input/Output Supervisor

When a channel program must be retried because of an error in its execution, SVC 15 is issued, usually by IBM-supplied error routines.

Execution of the EXCP macro instruction or SVC 15 causes an interruption. This and all other interruptions in the system are handled by the first level interruption handlers (FLIH).[3] After storing the program status word (PSW) and saving the contents of general registers, a FLIH routes control to the routine that performs the functions required by the interruption.

The I/O supervisor receives control from a FLIH after the following interruptions:

- Supervisor call (SVC) interruptions, caused when SVC 0 (EXCP) and SVC 15 are executed.
- I/O interruptions of all types.

An SVC interruption also occurs when SVC 16, 17, 24 or 33, respectively, for the PURGE, RESTORE, DEVTYPE or IOHALT macro instruction is executed. A FLIH gains con- | trol; however, if the required purge or restore routine is transient, it must be called into the SVC transient area before control is given to it.

Figure 3 shows the flow of control within the system and within the I/O supervisor from the time an SVC or I/O interruption occurs until the time the interruption is handled and the control program or a processing program is given control.

The mainline code of the I/O supervisor is assembled as a single object module at system generation (SYSGEN) time. Included are five modular areas (these are not load modules). Three of these areas are within the EXCP supervisor and two are within the I/O interruption supervisor. The size and function of each of these areas depends upon system configuration and their inclusion in the I/O supervisor is determined at SYSGEN time. Hence, the assembly listing produced as SYSGEN output is the only listing of I/O supervisor mainline code.

----

[1] Access method descriptions are contained in the <u>IBM System/360 Operating System: Supervisor and Data Management Services</u> publication, Form C28-6646.

[2] The necessary information is contained in the section "Execute Channel Program (EXCP) Macro Instruction" of the <u>System Programmer's Guide</u>.

[3] The FLIHs are described in the <u>IBM System/360 Operating System: Fixed-Task Supervisor, Program Logic Manual</u>, Form Y28-6612 (for PCP and MFT users) and the <u>MVT Supervisor, Program Logic Manual</u>, Form Y28-6659 (for MVT users).

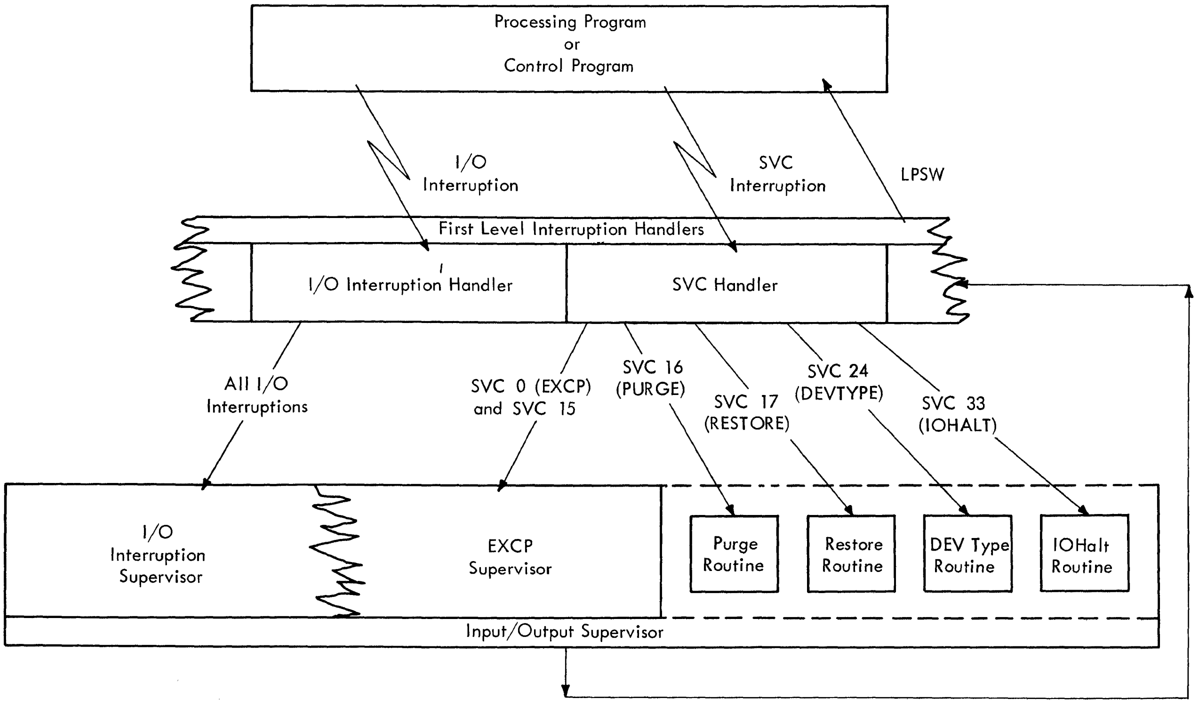


Figure 3. Control Flow Within the System and the Input/Output Supervisor

The modular areas, shown in Table 1, are:

- Test channel. A test channel module determines if a channel is available for the device that is to be used for an I/O request.
- Enqueue and dequeue. An enqueue module queues I/O requests that cannot be immediately started. After an I/O request has finished, a dequeue module removes the request from the queue.
- Start I/O. Start I/O modules provide device information and actions preceding the issuance of a start I/O instruction for the device.
- Channel search. When an I/O request has finished such that a channel becomes available, a channel search module determines the next I/O request to be started on that channel by searching queues constructed by enqueue modules.
- Trapcode. Trapcode modules establish information for a device after an I/O interruption associated with that device.

Table 1. Modular Areas Within the Input/Output Supervisor

Major Program Section	Modular Area	
	Name	Features
EXCP Supervisor	Test Channel	One module per logical channel; created at SYSGEN time.
	Enqueue and Dequeue	Two modules; created at SYSGEN time.
	Start I/O	One module per device class; selected at SYSGEN time.
I/O Interruption Supervisor	Channel Search	One module per physical channel; created at SYSGEN time.
	Trapcode	One module per device class; selected at SYSGEN time.

The load module names for the I/O supervisor are IEAAIH00 for PCP and MFT and IEAQFX00 for MVT. This load module includes the mainline code of the I/O supervisor, the five modular areas described above, and in addition, includes the dispatcher, the exit effector, the FLIHs, and the communication vector table

(CVT) of the task supervisor. The load module names for the various error routines and the purge and restore routines are summarized in Appendix A.

Routines that supplement the handling of I/O requests and interruptions are appendages to the I/O supervisor. (They do not alter the I/O supervisor program.) These appendages may be supplied by the user of the I/O supervisor.

Appendages are called into main storage by the open routine and are removed by the close routine or at the end of a job step.[1] When the I/O supervisor branches to an appendage, the current PSW has all interruptions except machine check interruptions masked as disabled; it indicates the supervisor state, and a protection key of zero.

There are five exits from the I/O supervisor to an appendage vector table; the table contains addresses, which may or may not route control to an appendage. (The appendage vector table is described in Section V: Data Extent Block.) If control is not routed to an appendage, the I/O supervisor retains control and continues normal processing.[2]

The five appendage exits are:

- End-of-extent. The provision for an end-of-extent appendage is included within the EXCP supervisor for direct access device I/O requests. The appendage is given control when the EXCP supervisor determines that execution of the request will violate the extent limits of the data set.

- Start I/O. The provision for a start I/O appendage is included within the EXCP supervisor for all I/O requests. The appendage receives control preceding the execution of the start I/O (SIO) instruction for the request.

- Program-controlled interruption. The provision for an appendage for a program-controlled interruption (PCI) is included within the I/O interruption supervisor. The PCI appendage returns control to the I/O interruption supervisor where normal processing continues.

- Channel end. The I/O interruption supervisor exits to a channel end appendage after a channel end interruption occurs, before indicating to the user how the I/O request completed. (The channel end appendage is also entered when the channel end interruption is accompanied by a wrong length indication or a unit exception, or both.)

- Abnormal end. The I/O interruption supervisor exits to an abnormal end appendage whenever there has been an error associated with the processing of an I/O request.

EXCP SUPERVISOR

The EXCP supervisor is the portion of the I/O supervisor that starts the channel programs requested by use of the EXCP macro instructions and SVC 15. It performs the same procedures when initiating the channel programs for both. Chart 01 shows the functional flow of the EXCP supervisor, and also of the I/O interruption supervisor. (All charts precede Appendix A.)

Before accepting a channel program for execution, the EXCP supervisor determines if:

- The control blocks for the I/O request are valid.

- The device for the I/O request is available.

- A physical channel for the device is available.

If the first of these conditions is not met, either a program check results or the task requesting the I/O activity is abnormally terminated. If either the device or the channel is not available, the I/O request is queued for subsequent processing.

When the required conditions are met, the EXCP supervisor issues an SIO instruction, which in turn initiates the commands in the channel program. The following conditions may result:

- The channel program may be accepted and the I/O operations successfully started.

- An error may occur during initiation of the channel program, in which case, control is given to the I/O interruption supervisor and the reason for the error is determined. If necessary, an error routine is scheduled.

[1] The OPEN and CLOSE routines are described in the publication IBM System/360 Operating System: Input/Output Support (OPEN/CLOSE/EOV), Program Logic Manual, Form Y28-6609.

[2] Additional information regarding the use of appendages can be found in the EXCP section of the System Programmer's Guide.

- The control unit for the device may signal busy. In this case, the channel program is not accepted and the I/O request is queued.

After the resulting condition is handled, control returns to the FLIH.

I/O INTERRUPTION SUPERVISOR

An I/O interruption occurs after signals from I/O devices have been generated (in response to channel program execution and operator action) and accepted by the central processing unit (CPU). The I/O interruption supervisor analyzes status bits in the channel status word (CSW), which is stored as a result of the I/O interruption, to determine the reasons for the I/O interruption.

The I/O interruption supervisor processes the following, when denoted by the CSW status bits:

- Attentions. The appropriate attention routine is entered.

- Status modifier situations. Status modifier conditions, such as a control unit busy condition that causes an I/O request to be queued, are handled.

- Control unit end, channel end, and device end interruptions. These interruptions indicate normal ending conditions and normal processing continues. Normal completions are indicated to the user.

- Busy conditions. When the selected I/O device or control unit is unavailable, the I/O request is queued.

- Unit check. A sense command is issued to obtain additional information regarding the unit check.

- Errors and unusual conditions. When errors, such as program, channel chaining, and protection checks, and unusual conditions, such as unit exception and wrong length indication, are detected, an error routine is scheduled. An uncorrectable error is indicated to the user as a permanent error condition.

- Program-controlled interruptions. A PCI appendage supplied by the user of the I/O supervisor is entered.

- Channel errors. If channel control check or interface control check occurs, three possible procedures exist: (1) If no recovery management options are included in the system, the wait state is entered immediately. (2) If either SER0, SER1, or MCH is in the system, the system environment is recorded before a possible wait state is entered. (3) If the channel-check handler (CCH) is in the system, an analysis is performed to prepare for possible retry by a device dependent error routine.

Note: If the system is generated to operate in the multiprocessing mode, both CCH and MCH must be specified as recovery options.

The completion of an I/O request is indicated by a channel end or a channel end and device end interruption. The I/O interruption supervisor indicates this completion to the user.

After the reason for the I/O interruption has been determined and the interruption processing completed, the I/O interruption supervisor determines the state of the channel upon which the interruption was received. Pending I/O interruptions are accepted and handled. If the channel is free, an attempt is made to restart it with a queued I/O request (one that was previously requested, but could not be started immediately). The appropriate queues are searched until an I/O request is started or until it is determined that there are no I/O requests that can be started on the free channel. Control returns to the FLIH.

The Error Routines

Standard IBM-supplied error routines are selected at SYSGEN time for each type of I/O device in the system. Error routines are transient and reside in the SVC library. However, the portions of the direct access device error routine that are necessary for the system residence device and for frequent unusual conditions (end-of-cylinder, head switching, alternate track procedures, etc.) reside in main storage. The names of the error routine load modules are in Appendix A.

Whenever an I/O interruption occurs and the I/O interruption supervisor detects an error that requires error recovery procedures, it determines whether or not an IBM-supplied error routine is to be used. An indication of the required error routine and of the channel program that developed the error are passed to the task supervisor's exit effector and an asynchronous exit is requested. The error routine must be scheduled asynchronously because immediate in-line processing of the error would hinder efficient channel scheduling.

The appropriate error routine is entered into the I/O supervisor transient area (providing the error routine is not resident) and, when scheduled, is given control. The error routine determines the type of error and, where possible, issues SVC 15 in an attempt to retry the channel program and to recover from the error. At completion of error processing (successful or unsuccessful), the error routine issues SVC 15 so that the I/O request can be properly terminated by the I/O interruption supervisor. The completion is indicated to the user and the error routine returns control to the task supervisor.

The Purge and Restore Routines

The SVC purge and restore routines are transient and are called into the SVC transient area whenever a PURGE or RESTORE macro instruction (SVC 16 and SVC 17, respectively) is issued. The load module name for the SVC purge routine is IGC0001F and for the SVC restore routine is IGC0001G. Purge may be a resident routine, if this option is selected at SYSGEN time.

The SVC purge routine removes active or queued I/O requests from the system. The SVC restore routine reschedules I/O requests that were previously purged.

There is also an I/O purge routine; the load module name is IGE0025E. This routine is used only by the I/O supervisor and should not be confused with the SVC purge routine, which is called by means of a system macro instruction. The I/O purge routine removes certain requests from the system whenever there has been a permanent error. It is discussed in Section II: Error Routine Interface.

The Control Blocks, Tables, and Queues

The I/O supervisor, as well as other portions of the control program, records the status of devices, data sets, and internal routines in control blocks, tables, and queues.

The control blocks supply information to the I/O supervisor regarding I/O requests and devices, and regarding the data sets required for the execution of the requests. The tables point to device- and channel-dependent modules and are used internally by the I/O supervisor. Queues are used by the I/O supervisor for I/O requests that cannot be started immediately. Section V contains detailed formats and explanations of the control blocks and tables.

CONTROL BLOCKS

Control blocks contain the information that concerns the status and activity of I/O devices, requests, and interruptions. The I/O supervisor is concerned with the following control blocks:

- Input/output block
- Data control block
- Data extent block
- Event control block
- Unit control block

These control blocks, and the pointers indicating their relationships, are shown in Figure 4.

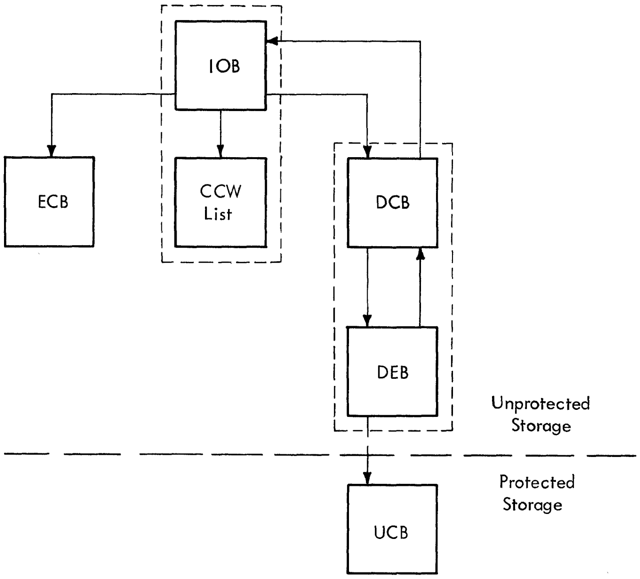


Figure 4. Control Blocks Used by the Input/Output Supervisor

Input/Output Block

An input/output block (IOB) is the communication medium between a routine that requests I/O operations and the I/O supervisor. The address of an IOB is passed to the I/O supervisor when an I/O request is given.

The contents of an IOB include:

- The address of the channel program, which consists of a group of channel command words (CCW), to be executed.
- An indication that the channel program is or is not related to other channel programs. (Related channel programs are all associated with the same data set. Each is dependent upon the successful completion of its preceding

related channel programs. If a permanent error occurs that is associated with a related channel program, the I/O purge routine purges the remaining related requests for the data set.)

- A storage area for the CSW, which is stored by the I/O supervisor at the completion of an I/O request.

- The initial seek address for a direct access device.

- The address of the data control block and the event control block.

- Flags which are used by the device dependent error recovery routines and which provide communication between these error routines and the I/O supervisor.

Data Control Block

A data control block (DCB) contains information that pertains to the current use of a data set and contains the address of a data extent block. A DCB provides the I/O supervisor with the status of error correction procedures for each data set and indicates whether or not IBM-supplied error routines are to be used for the data set.

Data Extent Block

A data extent block (DEB) is an extension of the information contained in a DCB. Each DEB is associated with a DCB, and the two blocks address each other. A DEB describes the extent of a data set; it indicates the storage medium, the location, and the boundaries of the data set, contains an appendage table address, and provides data protection and priority information.

Event Control Block

An event control block (ECB) is used by the I/O supervisor to indicate to the user when and how the execution of his channel program completed. A normal completion is determined by either a channel end or a channel end and device end interruption. The I/O supervisor indicates a request has completed, successfully or not, by posting the request complete by use of the post routine of the task supervisor.[1]

Unit Control Block

A unit control block (UCB) is the means by which the I/O supervisor notes and determines the activity and status of each I/O device attached to the system. A UCB represents the I/O device for the volume upon which a data set resides. It is addressed by a DEB.

There is one UCB for each I/O device or telecommunications line addressed in the system. One UCB per volume is required if a data set resides on two or more volumes that are mounted concurrently upon separate I/O devices. In this case, each of the UCBs is addressed by the same DEB.

The contents of the UCB include:

- The channel and unit address, which is used in the SIO instructions for the device.

- The type of device the UCB represents (e.g., IBM 2311 Disk Storage, IBM 2400 Magnetic Tape Unit, etc.).

- Flag bits, which indicate the current status of the associated I/O device. For example, the UCB busy bit may be set to indicate that the device is currently in operation. Other flag bits may be set to indicate UCB not ready, control unit busy, post, etc.

- An area for sense information, which details the conditions, such as a programming error or equipment malfunction, that cause a unit check interruption. When the I/O supervisor issues a sense command, up to six bytes of sense information from the control unit of the device is read into this area; the number of bytes of information that is meaningful varies with the device type.

- Reference fields, consisting of an indexing value or an address, that refer to tables. When the reference is an indexing value, the I/O supervisor adds it to the starting address of the pertinent table to obtain the address of the proper entry. When the reference is an address, it is of the proper entry.

- A work area for direct access device error recovery procedures.

TABLES

Tables contain pointers to control blocks and to device- and channel-dependent routines, modules, and queues. They also provide an area for device statistics. The

[1] For information concerning the post routine, refer to the Fixed-Task Supervisor, Program Logic Manual for PCP and MFT, and the MVT Supervisor, Program Logic Manual for MVT.

I/O supervisor contains the following tables:

- UCB lookup table
- Channel table
- Device table
- Statistics table
- Attention table
- Request element table
- Logical channel word table

These tables, and the pointers indicating their relationships, are shown in Figure 5.

UCB Lookup Table

The UCB lookup table contains the address of each UCB in the system and the values that are necessary to locate these UCB addresses within the table. The table is divided into channel values, control unit values, and UCB addresses.

After an I/O interruption occurs, the I/O supervisor calculates the address of the UCB that represents the I/O device associated with the interruption. The table values and the I/O address, which is stored in the PSW, are used to perform the calculation.

Channel Table

The channel table is used by the I/O supervisor to re-enable a channel for additional I/O interruptions and to locate the channel search module for the channel. The table contains one entry for each physical channel attached to the system.

Device Table

The device table is the means by which the I/O supervisor determines the proper device-dependent modules for a device. Each entry in this table contains the address of a start I/O module, an enqueue module, and a trapcode module.

Statistics Table

The statistics table contains one entry, which includes up to eight statistics counters, for each physical I/O device attached to the system. Each statistics counter is used by the IBM-supplied error routines to count the number of temporary read and write errors, equipment checks, interventions required, and other statistics that depend on the particular device type.

Whenever a statistics counter overflows or a permanent error occurs, the SER program's statistical data recording routine uses the statistics counters to update statistical data records. Statistical data records are expansions of the statistics counters. They reside on the system residence device, so that they are available for recall and analysis.

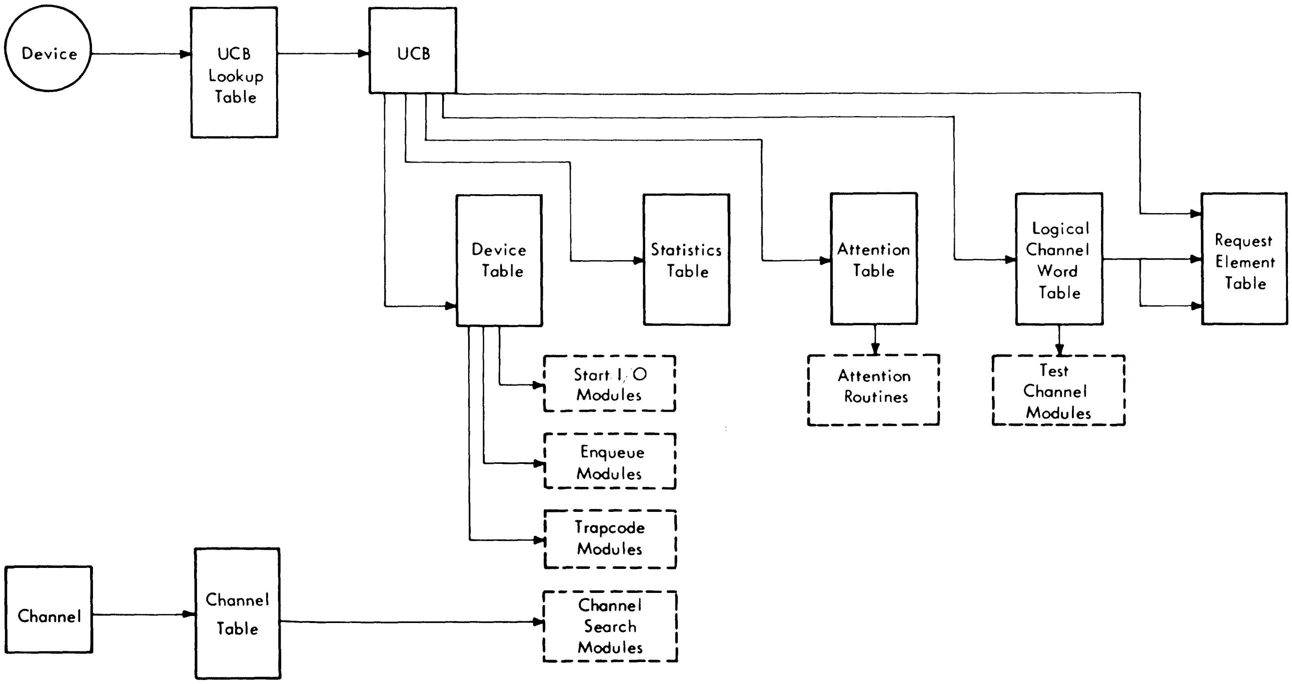


Figure 5. Tables Used by the Input/Output Supervisor

Attention Table

The attention table is constructed at SYSGEN time to contain the addresses of attention routines. Attention routines are specified by system routines to service attention interruptions.

There are two types of attention interruptions:

- Normal attention interruptions, which are indicated by the attention bit in the CSW status field.
- Unsolicited device end interruptions, which are caused by the transition from the not-ready to the ready state that occurs when the operator presses the ready button.

When a system routine desires that an attention or unsolicited device end interruption is to be handled, it places an indexing value into the UCB for the device. When the indexing value is added to the starting address of the attention table, the address of the appropriate entry within the table is obtained. The system routine must have placed into this entry, at SYSGEN time, the address of the attention routine for the device. Since the attention table is constructed at SYSGEN time, it is available only to routines that specify their attention routines at SYSGEN time.

When an attention interruption occurs, the I/O supervisor passes control to the attention routine. If execution of the attention routine is not desired, the system routine must place a zero into the UCB instead of an indexing value. The first entry in the attention table contains a branch back to the I/O supervisor; thus, when an attention interruption occurs, the starting address added to zero yields the address of the first entry in the table and the I/O supervisor regains control.

Request Element Table

The request element table[1] consists of a number of request elements that are used to represent I/O requests. The number of elements in the table is determined at SYSGEN time and remains fixed.

The I/O supervisor issues a request element to represent each I/O request given by means of the EXCP macro instruction. An element is not issued when an I/O request is retried by means of SVC 15. When the element is issued, the I/O supervisor stores information into it, which includes the addresses of the UCB, DEB, and IOB associated with the I/O request, the priority of the request, and the TCB ID for PCP and MFT or the TCB address for MVT.

Each request element can be in one of three states:

- Available
- Active
- Queued

An example of the linkage relationships of the available, active, and queued request elements within the request element table is presented in Appendix B.

Available Request Elements: Available request elements are not in current use; they are either unused or contain information that is no longer needed by the I/O supervisor. These request elements are issued to represent incoming I/O requests.

The available request elements are linked together by means of a "link field" that is contained in the first two bytes of each element. This linkage of available request elements forms a "freelist". Each element in the freelist points to the next element. Because other request elements in the request element table might be active or queued, members of the freelist are not necessarily contiguous within the table.

The first element in the freelist, referred to as "next available", is always the next request element issued by the I/O supervisor for an incoming I/O request. The "freelist pointer" contains the address of the next available request element. It resides within the I/O supervisor and is addressed by an entry in the communications vector table (CVT).[2]

The last element in the freelist is the last available request element and is identified by means of a "dummy address" in its 2-byte link field. The dummy address is a hexadecimal FFFF (i.e., has all bits on). When there are no available request elements, the freelist pointer contains the dummy address to indicate that the freelist is empty.

Active Request Elements: Active request elements correspond to I/O requests that have been started but have not yet been completed. The I/O supervisor indicates that a request element is active by placing

[1] In program listings the request element table is referred to as the 12 star (12*) table and its request elements as 12* elements.

[2] The CVT is described in IBM System/360 Operating System: System Control Blocks.

the address of the active request element into the UCB that represents the device being used for the I/O request. When an I/O interruption associated with the device occurs, the I/O supervisor finds the active request element, which is then used to find the various control blocks that are used during the processing of the I/O interruption.

Queued Request Elements: Queued request elements correspond to I/O requests that could not be started immediately. The elements belonging to the same queue are linked together by means of the link fields located in the first two bytes of the elements. The last element in each queue is denoted by means of the dummy address within its link field. During the queueing procedure, only the contents of the link fields in the request elements are changed.

Logical Channel Word Table

The logical channel word table contains a logical channel word entry that corresponds with each logical channel in the system. A multiplexor channel has one logical channel; a selector channel may have more than one. (For further information concerning logical channels, refer to Appendix C.)

A logical channel word designates the first and last request elements in the logical channel queue associated with a logical channel. If the queue is empty, the logical channel word contains in its first two bytes the dummy address of hexadecimal FFFF. A logical channel word also contains the address of a channel-dependent test channel module.

QUEUES

The I/O supervisor uses logical channel queues for I/O requests.

The logical channel queues consist of request elements that represent I/O requests for the execution of channel programs on either sequential or direct access devices.

Logical Channel Queues

I/O operations awaiting execution are held for subsequent processing in a logical channel queue. There is one logical channel queue associated with each logical channel in the system. (See Appendix C.) A multiplexor channel has one logical channel queue; a selector channel may have more than one. The last request element in each logical channel queue is identified by the dummy address in the link field of the element.

Queuing Procedures

The I/O supervisor effects linkage within the queues of request elements in the request element table by means of the first-in-first-out (FIFO) or priority queuing procedures.

In FIFO queueing, a request element is queued according to the order in which its corresponding I/O request is received (pushup queue). The queue is constructed and maintained so that the next request element to be retrieved is the oldest element in the queue.

Priority queuing is available with MFT and MVT. In priority queuing, either priority or FIFO is specified for each device. Requests for devices that have priority queuing are queued high to low according to the priority of the requesting task. Requests for devices that have FIFO queuing are placed lowest on the queue; a FIFO request has priority zero.

Section I: Execute Channel Program Supervisor

The EXCP supervisor receives control initially from the SVC FLIH after either an SVC 0 or an SVC 15 interruption. When the EXCP supervisor is entered, the current PSW has all interruptions except machine check interruptions masked as disabled, indicates supervisor and running states, and has the supervisor protection key of zero.

The EXCP macro instruction (SVC 0) is used to request normal I/O processing of a channel program. SVC 15 is used by the error routines to request that a channel program whose execution encountered an error be retried and also to indicate a permanent error or a successful retry. The I/O purge routine uses SVC 15 to return control to the I/O supervisor after related requests are purged.

The user of EXCP or SVC 15 is responsible for certain fields within the DCB and the IOB associated with the I/O request. Within the DCB, the user must ensure that:

- DCB exception bits set to 11, indicating a permanent error on the data set, are reset to 00. If this is not done, further related requests for that data set are posted as complete without execution.

- DCB exception bits set to 01, indicating error recovery procedures are in progress, remain at this setting.

- The tape block count remains unchanged until the IOB associated with the request has been posted complete.

Within the IOB, the user must ensure that:

- The block count increment amount for tape block counts is reset if the channel program indicated by the IOB changes.

  The field must be zero if block count updating is not desired.

  The field must be negative for a backspace.

  The field must be zero for a rewind; the DCB block count field must also be zero.

- The unrelated flag is on to indicate the I/O request is unrelated or is off to indicate the I/O request is related, and must be handled as such.

- The start address contains the address of the first CCW to be executed in the channel program.

- The channel program description indicates the type of chaining in the channel program, as follows:

00	No chaining
10	Data chaining
01	Command chaining
11	Data and command chaining

- The ECB address contains the address of the corresponding ECB.

- The DCB address contains the address of the DCB for the data set on which I/O processing is requested.

When the EXCP supervisor receives control, the address of the IOB that contains the information necessary for the I/O request must reside in general register 1. Processing of the I/O request then takes place.

After the operation of the EXCP supervisor is complete, control returns to the type 1 exit of the SVC FLIH, which determines whether control is to be given to the dispatcher or to the routine giving the I/O request.

Chart 02 shows control flow among the subroutines and the modules within the EXCP supervisor.

General Operation Procedure

For a normal I/O request, the EXCP validity check subroutine is entered directly from the SVC FLIH. This subroutine checks the validity of the control blocks associated with the I/O request. It also initializes certain IOB fields that are to be used in later processing.

For an error retry, the EXCP validity check subroutine is entered from the error EXCP routine. The error EXCP routine ensures that the IOB fields are not initialized, since their content is necessary for the error retry.

After the EXCP validity check subroutine is executed, the get request element subroutine assigns a request element to a normal I/O request. It stores address information required by the I/O supervisor into the next available request element.

Because a request element is assigned for all normal I/O requests, an additional request element is not issued if an error retry for a particular I/O request becomes necessary.

The flag bits within the UCB that denote the status of the I/O device are inspected to determine if the requested device is available for processing.

When the device is available, a test channel module determines if a physical channel in the device's logical channel is available. The correct test channel module is selected by means of the logical channel word table.

If conditions associated with the device prohibit processing or if the test channel module does not find an available physical channel, the request element is queued for subsequent processing of the I/O request. The appropriate enqueue module is selected by means of the device table entry for the particular device.

When a channel is free and the device is available, a test channel module selects the appropriate start I/O module for the device by means of the device table and gives control to it. The start I/O module performs device-dependent functions and branches to the start I/O subroutine, which in turn issues the SIO instruction for the I/O request.

The condition code and the CSW (if stored) that results from the execution of the SIO instruction indicate whether the channel program for the request has been started, whether the request element must be queued, or whether the SIO instruction for the I/O request should be reissued. The start I/O subroutine and the post-start I/O subroutine handle these results.

Once the operations for an I/O request are terminated (because of completion, error, or external action on the device), an I/O interruption occurs. Information concerning the operation of the I/O interruption supervisor is contained in Section II: I/O Interruption Supervisor.

Operation of the Subroutines and Modules

The following subroutines, routines, and modules are used to perform the functions of the EXCP supervisor (Chart 02 shows the flow of control between these routines):

- EXCP validity check subroutine
- Get request element subroutine
- Error EXCP routine
- Test channel modules
- Enqueue and dequeue modules
- Select subroutine
- Start I/O modules
- Start I/O subroutine
- Post-start I/O subroutine

EXCP VALIDITY CHECK SUBROUTINE

For both normal (SVC 0) and error retry (SVC 15) entries into the EXCP supervisor, the EXCP validity check subroutine validates the control blocks associated with the request. These control blocks are:

- Input/output block
- Data control block
- Data extent block
- Unit control block

The addresses of the control blocks are placed into general registers. During this process, the DEB and UCB identification fields are checked, the pointer to the DCB within the DEB is checked, and if applicable, the DEB protection tag is compared against the task control block (TCB) protection key. In MVT, the routine checks that the DEB is in system-protected storage (i.e., that its storage protection key is zero). If the subroutine finds that the control blocks are on improper boundaries, a program check occurs. On other errors, the ABTERM routine interface of the I/O interruption supervisor is entered and the task is abnormally terminated.

When the EXCP validity check subroutine is used for normal I/O processing, the following IOB fields associated with the last I/O request processed on the currently requested I/O device are reset:

- Error count
- Flags 1, 2, and 3
- CSW
- Condition code
- Sense
- ECB code

When the subroutine is used for error retry, these fields are not reset and control returns to the error EXCP routine.

RESIDENT SYSJOBQE ROUTINE

After the EXCP validity check subroutine is executed, the resident job queue routine (if selected at system generation) interrogates the request to determine if it refers to the resident portion of the job queue. If it does not, control is passed to the get request element subroutine. If it does refer to the resident portion of the job queue, the requested information is moved to its target location, an ECB is posted, and a type-1 exit is taken.

GET REQUEST ELEMENT SUBROUTINE

After a normal I/O request is validated, a request element is assigned to represent it. If an error retry for a particular I/O request becomes necessary, an additional request element is not required because one had been previously issued. Control block information is loaded into the request element, and UCB conditions, which determine if the request element must be queued, are tested.

If there is no available request element (i.e., the freelist pointer contains the dummy address of hexadecimal FFFF), the EXCP supervisor decrements the instruction address in the SVC old PSW by 2 and stores the new updated address in the SVC old PSW. For PCP and MFT, the EXCP supervisor executes an enable/disable instruction to allow I/O interruption to occur. If an interruption occurs, the I/O interruption supervisor processes the interruption and, if the interruption signifies completion of an I/O request, returns the associated request element to the freelist. The EXCP supervisor then gains control and returns to a type 1 exit which returns to the user's SVC 0 instruction, which gives EXCP control again.

The EXCP supervisor uses the request element that is returned to the freelist to represent the I/O request. However, if no request element can be returned to the freelist, or if no I/O interruption occurs, the EXCP supervisor repeats the enable/disable loop.

In MVT, no enable/disable instruction is executed. The TCB in question is marked non-dispatchable before the EXCP supervisor returns to a type 1 exit.

The EXCP supervisor places the following information in the next available request element:

- Address of the UCB, which represents the I/O device to be used for the requested operations.
- TCB identification, as noted in the TCB for the task requesting I/O operations. (In an operating system in which option 4 is included, the address of the TCB is also placed into the request element.)
- Address of the IOB, which indicates the channel program to be executed.
- Dispatching priority of the request, as noted in the DEB. (If no priority is specified, this field is not significant.)
- Address of the DEB, which indicates the location of the data set to be operated upon.

If the request is related and there is a permanent error, as indicated by the IOB unrelated flag and the DCB permanent error code, the request cannot be executed. (An error has occurred on a previous related I/O request for the data set.) In this case, the abnormal end appendage exit is taken. For a normal return from the appendage, the request is posted complete with the purge code and the request element is returned to the freelist. Control returns to the task supervisor.

UCB Condition Test: The conditions associated with the I/O device that may prohibit immediate processing are denoted by the following bits contained in the UCB:

- Control unit busy
- Device busy
- Not ready
- Disk arm seeking
- Error routine in control

If any of these bits are on, the request element is queued by an enqueue module for subsequent processing of the I/O request. If none are on, the device is available, is able to accept the request, and a test channel module is entered.

ERROR EXCP ROUTINE

The error EXCP routine directs the retry of I/O requests (and the results of each retry) to the I/O supervisor. The routine is entered from the SVC FLIH after the SVC 15 interruption. Chart 03 shows the flow of control in the error EXCP routine.

SVC 15 is provided to simplify error recovery procedures with respect to request elements; an additional request element need not be issued by the get request element subroutine.

IBM-supplied error routines issue SVC 15 to retry channel programs because of an error, and also issue it when the error has been corrected, when another retry is necessary, and when the error is permanent. (See Section III: IBM-Supplied Error Routines.)

The I/O purge routine issues SVC 15 when it has purged all related requests and the abnormal end appendage must be entered. (See Section II: Error Routine Interface.)

The resident portion of the direct access device error routine returns (it need not issue SVC 15) to the error EXCP routine: when an error is corrected; and when a request element must be queued for error recovery purposes. In the first case, the channel end appendage is entered; in the second, the channel restart subroutine of the I/O interruption supervisor is entered.

The only parameter that the user of SVC 15 must furnish is the address of the request element that represents the I/O request that is to be retried. The request element contains the address of the IOB associated with the I/O request and is loaded into general register 1 for entry to the EXCP validity check subroutine.

The EXCP validity check subroutine checks the validity of the control blocks pertinent to the request and loads their addresses into general registers. The EXCP validity check subroutine then returns to the error EXCP routine; the IOB fields are not reset.

When an error routine is in control (the IOB error correction indicator flag is set) and the request must be retried, the error EXCP routine tests the following UCB condition flags:

- Device busy
- Not ready
- Disk arm seeking
- Disk data transfer

The EXCP supervisor determines the result of the test, and the request element is either enqueued or a test channel module is entered.

When the error has been corrected and the error routine is no longer in control, the channel end appendage is entered. If the error has not been corrected, the error routine interface of the I/O interruption supervisor is entered, which indicates a permanent error condition and enters either the I/O purge routine or the abnormal end appendage, depending on whether or not the request is related.

TEST CHANNEL MODULES

When the device necessary for the I/O request is available, a test channel module determines whether or not a physical channel in the device's logical channel is available for device operation. If a physical channel is not available, control passes to an enqueue module and the request element is queued.

A selector channel test channel module exists for each logical channel in the system. One multiplexor channel test channel module exists for the logical channel associated with the multiplexor channel. In addition, common test channel code is used for both multiplexor and selector channels.

<u>Module Selection</u>: The test channel module associated with the device and its logical channel is selected by means of the logical channel word indexing value, which is contained in the UCB, and the logical channel word table. The proper logical channel word contains the address of the test channel module for the device.

Each test channel module (XCPCH1 -- XCPCH7) sets up the channel address and branches to common test channel code (XCPTCH or XCPTCH3).

Test Channel Operation

The test channel processing depends on the type of physical channel to which the requested device is connected.

<u>MULTIPLEXOR CHANNEL PROCESSING</u>: For the multiplexor channel, the common test channel code is entered at XCPTCH3. In this case, the common code does not issue a Test Channel instruction (TCH), since the condition code can indicate that the channel is free, although the needed subchannel is busy. The remaining processing, common to both multiplexor and selector channels, is described in "Common Processing."

<u>SELECTOR CHANNEL PROCESSING</u>: For selector channels, the common test channel code is entered at XCPTCH. It issues a Test Channel instruction for the channel path selected by the invoking test channel module (XCPCH2 -- XCPCH7). If the channel is busy, control is returned to the invoking test channel module in order to determine if there are additional paths to the device. If there are additional paths, the test channel module will select a new channel address and again branch to the common test channel code. This process continues until either all possible physical channels have been tested and none is free, or a free channel is found.

If all possible physical channels are busy or unavailable, the I/O request cannot be serviced. Control returns to either of two modules:

- An enqueue module to queue the request for future restart (see "Enqueue and Dequeue Modules").
- The channel restart subroutine to determine the next valid queued request that may be restarted (see "Channel Restart Subroutine").

<u>COMMON PROCESSING</u>: If at least one physical channel is not busy (or the multiplexor channel is selected), the common test chan-

nel code can continue (at XCPTCH3). It determines whether to branch to the appropriate device-dependent Start I/O module. The determination consists of testing the UCB intercept flag for an "intercept" condition, an abnormal condition remaining from a previous request for the same device.

If an intercept condition does not exist, the common test channel code gets the address of the appropriate Start I/O module and branches to the module. The common test channel code gets the address of the Start I/O module from the device table, by means of an index value in the device's unit control block. (For further information, see "Start I/O Modules".)

INTERCEPT PROCESSING: An intercept condition means that a previous I/O operation ended with an abnormal status (e.g., device end and unit check). Correction could not be attempted then because the event had already been posted complete. However, the abnormal status in the channel status word (CSW) and the sense bytes were saved in the device's unit control block (UCB).

The intercept processing code (XCPTCH5) handles the intercept condition as follows:

- Moves the saved CSW status bytes and the sense bytes from the UCB to the input/output block (IOB) associated with the current request. (The status and sense bytes are thus available for use by the error recovery routine that may be scheduled.)
- Sets the intercept code (hex. '7E') in the ECB code field of the IOB. The code (changed to hex. '44') will be posted in the event control block for the current request, if the intercept condition proves to be uncorrectable.
- Branches to the dequeue module (XCPPDQ) to remove the request element from the request queues. Control is then given sequentially to the user abnormal end appendage and the error routine interface (INTERR). If the user has selected the use of IBM error recovery procedures, the request element is used to schedule an error recovery routine. Otherwise, the request is posted complete with permanent error.

ENQUEUE AND DEQUEUE MODULES

If the channel program for an I/O request cannot be immediately started because a device or a physical channel is unavailable, the request element that represents the I/O request is queued. The queuing procedure used depends on the type of enqueuing specified by the user at SYSGEN time. FIFO and priority queuing procedures are available.

Module Selection: The address of the enqueue module is contained in the device table entry for the device class of the particular device. The proper entry in the table is found by use of the device table indexing value contained in the UCB that represents the device.

The dequeue module is entered by a direct branch.

Normal Enqueue Module

The normal enqueue module queues request elements in a FIFO arrangement. This module places the address of the request element that represents an I/O request into the link field of the last request element in the pertinent logical channel queue. This request element becomes the "last" element in the queue and the dummy address is placed into its link field. The previous last request element is then updated to point at the new last request element.

Priority Enqueue Module

At SYSGEN time, the user specifies priority queuing for certain devices. The request elements that represent the I/O requests are queued in an order that maintains the priority specified in the DEB for the pertinent task.

Priority queuing is effected by searching the logical channel queue until the priority point at which the request element is to be queued is found. The address of the request element is placed into the link field of the next highest or equal priority request element in the queue. The highest priorities are at the top of the queue. Within a given priority, requests are in a FIFO arrangement.

Dequeue Module

The dequeue module is used by the I/O supervisor to dequeue request elements from queues maintained in FIFO and priority arrangements. A search is made of the queue until the request element to be dequeued is located. The element is unlinked and the queue is relinked by means of the 2-byte link field.

START I/O MODULES

When a device and channel are both available for an I/O request, a start I/O module is entered. It establishes the prerequisite device-dependent commands (set mode command for tape, set file mask command for direct access devices, etc.), the CCW address fields for the channel address word (CAW), and the CCWs necessary for the operation of an I/O device. The start I/O subroutine is then entered to issue the SIO instruction.

Module Selection: There is one start I/O module for each device class in the system. The address of the appropriate start I/O module is obtained by the common test channel code. It adds the device table indexing value contained in the UCB that represents the device to the starting address of the device table. The entry in the device table at the resultant address contains the start I/O module address.

Graphics, Unit Record, and Telecommunications Start I/O Module

The graphics, unit record and telecommunications start I/O module is used for all I/O requests on unit record equipment and telecommunications devices. It loads a general register with either the IOB start or the IOB restart address field, depending on the setting of the IOB start/restart flag. Control is then given to the start I/O subroutine where the CAW is loaded with the contents of the register.

However, when an immediate operation is required by an error routine, control goes directly to the start I/O subroutine.

Tape Start I/O Module

The tape start I/O module is used for all I/O requests on the 2400 tape series, including the 2420 Model 7. This module prepares the following command chained CCWs, contained within I/O supervisor storage, for execution by the start I/O subroutine:

- First CCW. A set mode command, which is specified in the modifier byte of the DEB extent by the open routine, is placed in the command code field. This command sets the mode (parity, density, etc.) of operations on the tape unit.
- Second CCW. The command code field contains a transfer-in-channel (TIC) command. The data address field is loaded with the IOB start or restart address.

Control passes to the start I/O subroutine.

Tape Unit Positioning: When repositioning (backspacing, forward spacing, erasing) is necessary for error recovery procedures, control passes immediately to the start I/O subroutine where a CCW is loaded with the repositioning command code, and chained to a NOP CCW. The code is specified in the IOB modifier byte by the 2400 tape series error routine.

After a data check during a read or read-backwards operation, the I/O Supervisor uses the following command-chained CCWs. These CCWs prepare for an error recovery attempt in the opposite direction:

- First CCW. A set mode command, which is specified in the modifier byte of the DEB extent by the open routine, is placed in the command code field. This command sets the mode (parity, density, etc.) of operations on the tape unit.
- Second CCW. The command code field contains a track-in-error (TIE) command to send the second sense byte to the control unit.
- Third CCW. The command code field contains a transfer-in-channel (TIC) command. The data address field is loaded with the UCB RORCCW address. The RORCCW is used for read opposite recovery.

Cyclic Redundancy Check Correction: When cyclic redundancy check (CRC)[1] correction is required, the tape start I/O module loads the first CCW command code field with a "track in error" command code (hexadecimal 1B). It also places the second sense byte, which contains the track-in-error information, into the CCW for later transfer to the tape adapter unit (TAU).

Direct Access Start I/O Module

The direct access start I/O module is used for all I/O requests on direct access devices. The extent limits of the data set are checked; if they will be violated, the end-of-extent appendage exit is taken. If the requested device has movable access arms, a stand alone seek and a subsequent start data transfer seek are used to start the channel programs. If the request is for a device such as the 2301 drum storage unit (which has a fixed access mechanism) the stand alone seek is not necessary, and only the start data transfer seek is issued.

If the request is for a shared direct access device (indicated by bit 2 of the optional features byte of the UCB device class field), the module issues a reserve command with the stand-alone seek to prevent access arm movement caused by another CPU before the subsequent start data transfer seek is issued.

Stand Alone Seek: The stand alone seek is a preliminary seek command that is necessary to position the access arm before a

[1] Information regarding a CRC error is contained in the IBM 2400 and 2816 Model I: Component Description publication, Form A22-6866.

channel program for a direct access device can be initiated. The stand alone seek starts the access arm seeking for one I/O request while allowing other I/O operations on the same physical channel.

The stand alone seek is issued by means of a stand alone CCW contained within I/O supervisor storage. The module loads the CCW with the seek command code and a pointer to the seek address provided by the user within the IOB. The file mask at this time allows all seeks, having been automatically reset following the last operation.

Before issuing the stand alone seek, the direct access start I/O module ensures that all user-specified seek addresses are within the extent limits indicated by the DEB that is associated with the data set to be operated upon. The module stores the IOB seek address into the UCB for later start data transfer seek procedures and then indexes the extent area of the DEB with the M EXTENT field of the IOB to locate the correct extent.

The module checks the IOB seek address to determine whether or not it is within the limits of that extent. If not, an end-of-extent appendage is entered.

If the seek address is within the extent limits, the direct access start I/O module sets the disk arm seeking flag in the UCB to indicate that the direct access device is involved in a seek operation and loads the address of the stand alone seek CCW in a general register. Control passes to the start I/O subroutine, which stores the contents of the general register in the CAW, enters the start I/O appendage, and then issues the stand alone seek to position the access arm for the direct access device.

If a CSW is not stored upon return from the start I/O subroutine, the module effects a loop with a test I/O (TIO) instruction until the CSW is stored. The stored CSW indicates that channel end has occurred and that the seek address has been transferred to the head register of the control unit. Then the module tests for any abnormal conditions received as a result of the TIO instruction. If no abnormal conditions are indicated, the status is stored in the associated IOB.

The module tests to see if the access arm has reached its destination (device end). If the seek was not completed, the module either branches to the enqueue routine if the module was entered via the EXCP subroutine, or gets the next request via the channel search module if the module was entered via the channel restart procedure. If the seek was completed, the UCB busy and post flags are turned off indicating that the device has reached its destination and that the data transfer operation can be started.

If the module was entered via the EXCP subroutine, the data transfer portion of the channel program is started. Otherwise, the module branches to the channel search subroutine to obtain the next request in the queue.

The address of the request element for the I/O request was placed into the UCB at SIO time.

<u>Start Data Transfer Seek</u>: Device end for a stand alone seek indicates that the access arm is in position for data transfer operations and that the channel program may be initiated. The start data transfer seek is issued when the channel search module within the I/O interruption supervisor determines that the associated physical channel and direct access device are available. The start data transfer seek initiates the channel program.

At system generation time, three CCWs are provided for each channel which contains any direct access devices. The direct access start I/O module finds the appropriate set of command chained CCWs by obtaining pointers to the corresponding channel search code.

To issue the start data transfer seek, the direct access start I/O module first determines if the device is shared. If it is, the module checks the reserve count in the UCB. If the reserve count is zero, the module sets up a device release command as the first CCW. The release command frees the device for use by another CPU as soon as the data transfer is completed. The module then prepares the following command-chained CCWs. (If the device is not shared, or if the shared device's UCB reserve count is not zero, the following CCWs are the only ones used.)

- <u>First CCW</u>. The command code field contains a seek command. The data address field contains the address of the seek address that was stored in the UCB during the stand alone seek operation.

  When the command is executed, the seek address overlays the control unit head register. The head register must be reinitialized to the proper seek address because the seek address from the stand alone seek associated with this channel program might have been destroyed by subsequent seek commands to other direct access devices on that control unit.

- Second CCW. The set file mask specified in the DEB, which describes the types of write and seek commands that can be initiated on this data set, is loaded into this CCW.

- Third CCW. The command code field contains a TIC command. The data address field is loaded with either the IOB start or the IOB restart field, depending on the setting of the IOB start/restart flag.

  When the command is executed, control passes either to the user channel program or to an error recovery channel program.

The direct access start I/O module sets the disk data transfer flag in the UCB, and branches to the start I/O subroutine, which initiates the CCW execution. The results of channel program execution are handled by the post-start I/O subroutine.

End-of-Extent Appendage

Entry into the end-of-extent appendage is made by means of the appendage vector table. Upon return, the appendage specifies one of the following:

- To disregard the error and try again.
- To skip execution of the request and have it posted complete.
- To indicate a permanent error, enter the abnormal end appendage, and then post the request complete.

START I/O SUBROUTINE

The start I/O subroutine issues all SIO instructions for the start I/O modules.

When tape and unit record start I/O modules require immediate operations or unit repositioning, the start I/O subroutine places the operation code contained in the IOB modifier byte into a stand alone CCW for execution. It places the address of that CCW into the CAW.

The start I/O appendage exit is then taken before the SIO instruction is issued. This exit is not taken, however, when a start data transfer seek is to be issued (unless it is a drum type device) or when an error routine is in control. If the protection feature is specified at system generation time, the protection tag is moved into the CAW location from the DEB for MFT and from the TCB for MVT.

Before the start I/O instruction is issued, the start I/O subroutine stores the channel and unit address and the request element into the associated UCB. The start I/O subroutine issues the SIO instruction directly.

Execution of the SIO Instruction: The start I/O subroutine issues the SIO instruction after loading the CAW with the address of the starting CCW and after loading the SIO instruction with the unit address contained in the UCB.

After execution of the SIO, the SIO subroutine checks for condition code 2 which indicates that the path to the device is busy. If the path is busy the routine stores the control unit busy indication in the CSW. The PSW condition code, the instruction length code, and program mask from the PSW are stored in the IOB. The UCB associated with the SIO is stored in a channel table to be used at interruption time. Also, the UCB busy and post flags are set.

The SIO subroutine then tests for condition code 3 (device not operational). If the condition code is 3, a branch is made to a subroutine to simulate intervention required; otherwise control is passed to the post SIO subroutine.

Start I/O Appendage

A start I/O appendage is given control preceding the execution of the SIO instruction for an I/O request. For direct access devices, the appendage receives control preceding the execution of the SIO instruction for the stand alone seek. Entrance to a start I/O appendage is by use of the appendage vector table. Its address is placed in the DEB by the open routine.

The start I/O appendage returns control to the EXCP supervisor with the indication to continue with normal processing or to skip processing of the request. If the result is skip, control is given to the dequeue module and the I/O request is not posted complete.

POST-START I/O SUBROUTINE

The post-start I/O subroutine determines the results of the SIO instructions issued by the start I/O subroutine. It analyzes the condition code and, if the CSW is stored, the CSW status bits.

SIO Instruction Accepted: A condition code setting of 0 indicates the acceptance of the SIO instruction. This implies that the channel program is successfully started.

The request element is dequeued from the logical channel queue and its address is placed in the UCB.

CSW Stored: When the CSW is stored, indicated by a condition code setting of 1, the CSW status bits may indicate the following:

- Channel errors. When a channel error occurs, control goes either to the SER interface or the CCH interface.
- Busy. The following flags are set to zero in UCBFLAG:

  UCB busy
  Post
  Disk data transfer
  Disk arm seeking

  If busy is accompanied by control unit end, the I/O request is retried. If not, the UCB is marked with the control unit busy flag. If the status modifier bit in the CSW is on, the post SIO subroutine returns to the test channel subroutine. If the status modifier bit is off (busy alone is indicated), the post SIO subroutine returns to enqueue or to get the next RQE.
- Channel end. The operation was an immediate command. The I/O request is posted complete and, if device end is also on, the UCB busy bit is cleared.
- Program or Protection Check. The abnormal end appendage is entered and, if the check is not reset, the task is abnormally terminated. The I/O request is posted complete with an error code ('41').
- Unit check. The sense subroutine issues a sense command, the abnormal end appendage is entered, and then the error routine interface is entered.
- Attention. An attention interruption for the device may be brought in. The attention routine interface is given control.

Condition Code 3 -- Device not Available: A condition code 3 implies that the device is not available or that a path to the device is not operative. If the device has more than one path, and if this is the first time this inoperative path has been encountered, an error routine is entered to issue an informational message to the operator. If the inoperative path has been encountered previously, the test channel module is entered to check for another path to the device.

MULTIPROCESSING EXTENSIONS

The I/O Supervisor is extended to implement multiprocessing, if the M65MP option is selected at system generation. In multiprocessing, all I/O devices except the consoles and teleprocessing devices are accessible from each CPU through its own channels. If the CPU originating an I/O operation finds that its own channel is not available, it tests the availability of the corresponding channel of the other CPU (all channel-control unit-device addresses are the same for both CPUs). If the other CPU's channel is free, the originating CPU issues a "shoulder tap," causing an external interruption on the receiving CPU, which then initiates the I/O operation.

Tables and Work Areas

The Unit Control Block (UCB) is extended by a full word for multiprocessing. (See Figure 22, Section 5.) The first byte of this word contains flags indicating whether both CPUs or only one has access to the associated device, and which CPU started the current I/O operation, if one is in progress. The extension is located "in front of" the UCB, but does not affect the displacement of other UCB fields. It is addressed by subtracting four from the UCB address.

Multiprocessing requires a new Channel Availability Table in the Prefixed Storage Area (PSA) of each CPU. (See Figure 9, Section 5.) A channel is marked unavailable in this table when:

- SIO is issued. This condition is cleared by the I/O Interruption Supervisor when the I/O operation completes.
- The Nucleus Initialization Program determines that the channel is not operational or not configured in the system.
- A VARY CHANNEL or VARY CPU command from the operator places the channel offline.

Extended Test Channel Logic

The test channel common routine (XCPTCH) of the EXCP Supervisor, under multiprocessing, makes a second test after finding that the requested physical channel is busy or not available. The test channel common routine determines if the corresponding channel of the other CPU is free by testing the Channel Availability Table in the other CPU's Prefixed Storage Area. If the table shows the channel is free, and the UCB extension flags indicate the other CPU has access to the device, the shoulder tap interface subroutine (XCPSTI) is given control. The subroutine sets a flag in the shoulder tap control word (STMASK), then branches to the shoulder tap routine (SHOLDTAP). The shoulder tap routine issues the Write Direct instruction to notify the other CPU. On return from the shoulder tap interface subroutine, the I/O request element is queued to that channel, where the other CPU will find it after the supervisor is unlocked by the requesting CPU (only one CPU executes the I/O Supervisor at a time).

If neither CPU has an available channel, the request element is queued pending an interruption on either CPU.

Multiprocessing Subroutines

The following subroutines (with entry points in parentheses) are included in the I/O Supervisor for the multiprocessing option:

- Shoulder Tap Interface -- Sending. This subroutine sets a flag in the shoulder tap control word in the PSA of the sending CPU that indicates to the receiving CPU which channel an I/O operation is requested for. The subroutine branches to a common system routine to execute the Write Direct instruction causing an external interruption on the other CPU.

- Shoulder Tap Interface -- Receiving (IECISHTP). This routine receives control from the external FLIH after an I/O shoulder tap. It tests the channel flags in the shoulder tap control word of the sending CPU's PSA, and for each channel flagged, goes to the channel restart routine of the I/O Interruption Supervisor to start the request, which has been placed on the channel queue by the sending CPU.

- VARY Interface (IECTIOMP and IECTCHMP). This subroutine issues the Test I/O and Test Channel instructions for the VARY command handling routines.

- Accessability Test. This code is part of the common test channel code (XCPTCH2). It tests the device accessability flags of the UCB extension to ensure that a CPU about to start an I/O operation has access to the device. A CPU does not have access to the other CPU's console, and only one CPU has access to teleprocessing devices.

- Halt I/O Check (IECMPHIO). An I/O operation can only be halted by the CPU that initiated it. The PSA of each CPU contains a unique identifier which is placed in the UCB extension whenever SIO is issued for a device. When a Halt I/O is requested, this routine compares the identifier of the executing CPU to the identifier in the UCB extension. If they are the same, control is returned to the resident HIO routine. If different, flags are set in the shoulder tap control word and the UCB extension, and a shoulder tap is performed.

- Shoulder Tap Halt I/O (HLTIOUNT). This routine receives control following a shoulder tap for Halt I/O. It searches for the flags set by the Halt I/O Check routine, and passes control to the resident HIO routine.

Section II: I/O Interruption Supervisor

When an I/O interruption occurs, the I/O First Level Interruption Handler (FLIH) passes control to the I/O interruption supervisor. The I/O interruption occurs either in response to channel program execution or because of operator intervention. The conditions that cause the I/O interruption are indicated in the CSW.

When the I/O interruption supervisor receives control, the current PSW is in the supervisor and running states, has the supervisor protection key of zero, and has all interruptions, except machine check interruptions, masked as disabled.

After operation of the I/O interruption supervisor is complete, control returns to the I/O FLIH, which branches to the dispatcher (IEA0DS). The dispatcher gives control to either the current task or another ready task.

Charts 04 through 12 show the control flow among the subroutines and modules within the I/O interruption supervisor.

General Operating Procedure

The I/O interruption supervisor finds the address of the UCB that represents the interrupted device by picking up the last UCB started on this channel which was stored at SIO time or by means of the UCB lookup routine. This routine uses an algorithm with the values contained in the UCB lookup table to find the UCB, which in turn contains the address of the active request element for the interrupted I/O request. (In cases where the request element was already returned to the freelist, such as when an I/O interruption occurs because of an error after channel end, there is no active request element.) From the UCB and the request element, the pertinent control blocks, tables, and modules are located.

The I/O interruption supervisor analyzes the CSW status bits to determine the reason for the I/O interruption and then proceeds to service it. For example, when channel failures occur (indicated by channel control check or interface control check bits), either the SER interface or the CCH interface is entered. When the PCI bit is set, a PCI appendage is entered. When the attention bit is set, the attention routine interface is entered and control is given to an attention routine. For a unit check condition, the sense subroutine is entered; this subroutine issues a sense command and reads additional information regarding the I/O interruption into the sense field of the UCB. If an error or unusual condition occurs (the I/O request did not complete normally), the error routine interface is entered; this routine determines whether or not an IBM-supplied error routine should be scheduled for the condition.

When the I/O request completes normally (indicated by an interruption caused by channel and/or device end), a trapcode module is entered. The device table is used to select the correct trapcode module for the type of device. For direct access devices, the trapcode module determines if the stand alone seek is completed. For tape, the trapcode module updates the DCB block count. For other devices, a trapcode module only facilitates device independence.

The I/O request is posted complete by the task supervisor's post routine, which is entered by means of the I/O interruption supervisor's post routine interface. The type of completion (error, normal, etc.) is specified in the ECB. When the I/O request is completed and the channel is free, the I/O interruption supervisor attempts to restart the channel.

A channel search module for a particular channel is selected by means of the channel address and the channel table. The module searches the logical channel queues associated with the channel for a request element that represents an I/O request that can be started on the channel. The channel search module is re-entered until a request that can be started is found or until it is determined that there are no such requests available.

The I/O FLIH regains control when the channel is either restarted (by the start I/O routines described in Section I) or it is determined from the queues that there is no request that can be started.

When the channel has been restarted after an interruption, the I/O supervisor enables all channels to test for stacked interruptions. If none are pending the I/O supervisor again disables the channels and returns to the dispatcher.

Operation of the Subroutines and Modules

The following subroutines, routines, and modules are used to perform the functions of the I/O interruption supervisor. Major entry points are indicated within parentheses:

- UCB lookup routine (INT001)
- I/O interruption analysis (IECINT)
- Trapcode modules
- Channel restart subroutine (INT030)
- Channel search modules
- Sense subroutine (INTSEN)
- Interfaces
- I/O purge routine (PRGCOMA)
- Channel-check handler (CATNIP)

UCB LOOKUP ROUTINE

The UCB lookup routine locates the address of the UCB for the device associated with the I/O interruption by use of the 11-bit I/O address, stored in the I/O old PSW, in conjunction with the values in the UCB lookup table. The I/O address consists of the channel, control unit, and device addresses associated with the I/O interruption. The UCB lookup table, consisting of the channel, control unit, and UCB address list portions, is described in detail in Section V: UCB Lookup Table.

The I/O interruption supervisor locates the UCB for all types of I/O interruptions except for catastrophic errors and for the setting of the CSW control unit end bit alone. A figure showing the algorithm used to obtain the address of the UCB can be found in Section V: UCB Lookup Table.

The UCB contains the address of the active request element (representing the I/O request for which the I/O interruption occurred), which in turn contains the addresses of the associated IOB and DEB. The DEB contains the address of the DCB. The addresses of these control blocks are placed into general registers, which are not changed during the processing of the I/O interruption.

If the I/O interruption is such that the I/O request has been previously posted complete, there is no active request element; the element has already been returned to the freelist. This occurs for devices that have channel end and device end status presented separately.

I/O INTERRUPTION ANALYSIS

The I/O interruption supervisor analyzes the CSW status bits, stored as a result of an I/O interruption, to determine the cause of the interruption and the processing procedure that must be followed. The CSW status bits are analyzed in the order in which they are described. Charts 04 and 05 show the details of the I/O interruption analysis.

Channel Data Check, Channel Control Check, Interface Control Check Bits

The CSW status bits may indicate the presence of a channel data check, a channel control check, or an interface control check. These are considered catastrophic unless the channel-check handler routine (which attempts recovery from channel control and interface control checks) is part of the system. The possible actions are: (1) the system is placed in a wait state, (2) control is passed to the SER interface, or (3) control is passed to the CCH interface, depending upon the recovery option selected at system generation time.

Catastrophic errors are checked, and thus detected, upon entry into the I/O interruption supervisor. If such an error exists, the other CSW status conditions may not be accurate.

Control Unit End Bit

The setting of the control unit end bit alone, (without a channel end, device end, or unit check bit setting) is a response to the "control unit busy" condition (busy with status modifier) that is presented when the control unit is interrogated while executing an operation. It may occur when a control unit that is shared by two or more I/O devices or channels becomes free. The channel restart subroutine is given control.

PCI Bit and PCI Appendage

If the PCI bit is set, the PCI appendage supplied by the user of the I/O supervisor is entered. Because a PCI condition may ride in with other I/O interruptions, channel status conditions (program, protection, chaining checks) as well as unit status conditions are inspected upon return from the appendage. If no additional status conditions are present, control returns to the I/O FLIH. If status conditions are present, analysis is continued and the conditions are handled.

Channel End Bit and Channel End Appendage

The channel end bit indicates the completion of an I/O request. The I/O interruption supervisor moves the CSW status to the IOB for subsequent examination by the user of the I/O supervisor. The trap code module is entered and then the channel end appendage exit is taken.

The appendage returns control to the I/O interruption supervisor and specifies one of the following:

- Post request complete, and return request element to the freelist (normal return)
- Do not post complete, but return request element to the freelist
- Reschedule the request for another try
- Ignore the interruption pending an asynchronous user routine

Device End Bit

Device end is caused either by the completion of an I/O operation at the device or by the manual change of the device from the not-ready to the ready state. The following conditions may be present with device end:

- Device end with the UCB busy flag set, which indicates that an I/O device has completed its most current operation. The UCB busy flag is turned off and the operational status of the associated channel is interrogated.
- Device end alone (without the UCB busy or post flags set), which indicates that the I/O interruption is an unsolicited device end caused by the transition from the not-ready to the ready state. The attention routine interface is entered.
- Device end with the UCB post flag set, but with no channel end bit set, which indicates either that channel end has been suppressed by command chaining or that an error occurred at channel end. The UCB busy bit is turned off, and the CSW status bytes are moved to the IOB for the user. If there are any errors, the sense command is issued and an error routine is scheduled. If no errors are indicated, the device dependent trap code module is entered, followed by the channel end appendage.
- Device end with the UCB busy flag set, with the UCB post flag not set, and with a unit check or unit exception status, which indicates that an error occurred at device end. The UCB intercept flag is set to indicate that the error must be inspected at the next I/O request for the device.

Attention Bit

When the attention bit is set, the attention routine interface is entered. The attention routine interface is discussed later in this section.

Unit Check Bit and Abnormal End Appendage

When the unit check bit is set, the sense subroutine is entered so that additional information about the I/O interruption may be obtained. The unit check bit usually occurs with channel or device end bit settings (or both) or during the initial selection of the device.

The I/O interruption supervisor sets the IOB exception flag to indicate an error may be present, and takes the abnormal end appendage exit. The appendage returns to the I/O interruption supervisor and specifies one of the following:

- Continue normal processing
- Skip further processing of request
- Reschedule request
- Enter IBM-supplied error routine

When an error routine is entered, error correction is attempted; however, if the error routine determines that the error is permanent, the abnormal end appendage is re-entered. The preceding returns are applicable.

Unit Exception and Incorrect Length Bits

The unit exception and incorrect length bits indicate that a device-dependent condition is present. The IOB exception flag is set to indicate that an error may be present, and the channel end appendage is entered. When control returns to the I/O interruption supervisor, the error routine interface is entered, unless the appendage specifies otherwise. On return from the error routine, the abnormal end appendage is entered.

Program Check, Protection Check, and Chaining Check Bits

When the chaining check status bits are set, the abnormal end appendage exit is taken. Upon return, the error routine interface is entered to schedule an error routine that will inspect the type of status bit indicated, unless the appendage specifies otherwise.

When the program check or the protection check status bits are set, the abnormal end appendage is entered. If the appendage does not change the setting of the bits and specifies to continue normal processing, an error routine is scheduled. The error routine sets the permanent error condition. After control is returned to the I/O supervisor, the abnormal end appendage is entered again. On normal return from this appendage, the request is posted.

Status Modifier and Busy Bits

The status modifier and the busy bits indicate a control unit busy condition. (See "Control Unit End Bit.") The UCB is located by means of the UCB lookup routine and the UCB control unit busy flag is set.

TRAPCODE MODULES

A trapcode module, or the capability for it, exists for each device class. The trapcode modules provide device-dependent functions for channel end and device end interruptions. Upon return, either the channel status is interrogated or the channel end appendage exit is taken, depending on the settings of the UCB post flag.

Module Selection: The appropriate trapcode module is selected by use of the device table, which contains the address of the trapcode module to be used with a particular device class.

Unit Record and Telecommunications Trapcode Modules

The unit record and telecommunications trapcode modules perform no functions. A branch is made back to the I/O interruption analysis.

Tape Trapcode Module

The tape trapcode module updates the block count in the DCB using the block count increment amount specified in the IOB.

Direct Access Trapcode Module

The direct access trapcode module determines whether or not a seek is complete.

At the completion of a stand alone seek, indicated by a device end interruption, the UCB disk arm seeking flag on, and the UCB data transfer flag off, the direct access trap code module enters the channel restart module for a possible start of the data transfer portion of the channel program.

If a seek end has not occurred, the module returns directly to the I/O interruption analysis. The "both on" or "both off" settings of the disk arm seeking and disk data transfer flags in the UCB for the interrupted device indicate that there is no seek end.

Device End Post Trapcode Module

The device end post trapcode module guarantees a device end posting environment for graphic devices.

CHANNEL RESTART SUBROUTINE

The channel restart subroutine attempts to restart the channel after an analysis of the I/O interruption. A flag is set marking the channel to be restarted, and I/O interruptions are enabled. If an interruption occurs, the I/O interruption supervisor will receive control from the I/O first-level interruption handler. If no interruption occurs, I/O interruptions are disabled and channel restart continues.

A channel search module provides the channel restart subroutine with the address of the logical channel. To provide for overlapped seek and data transfer operations (on separate devices), the logical channel is searched for all requests which require seeks. Then it is searched again for a data transfer operation. When direct access devices are attached to this channel, a common channel search module is used to search the logical channel queue for direct access devices. When a request element for a seekable device is found in the queue, the status of the device is tested. If the device is not available, the module searches for the next request element for a seekable device. If the device is available, a test channel module is entered to determine if the channel is available.

A test channel module is not entered if the request is "related" and if a permanent error is indicated in the associated data control block (DCB). A related request is one issued for the same data set as a previous request. The related request is started only if the previous request has successfully completed. The request is recognized as related if the "unrelated" flag in the associated input/output block (IOB) is not set.[1] If the request is related, the channel search module tests the exception flag of the associated DCB for a permanent error condition.[2] If the DCB indicates a permanent error, the channel search module starts the following sequence:

- Places a "purged request" code (X'48') in the event control block code field (offset 04) of the associated input/output block. This code will eventually be used by the post routine interface (XCPPST) when it prepares to post an event control block.

[1] The "unrelated" flag is bit 6 of the FLAGS1 field (offset zero) in the IOB. For this test, the FLAGS1 field (IOBFL1) should not be hex. '02'.

[2] The DCB exception flag, when set, is denoted by hex. 'C0' at offset '2C' in the DCB.

- Branches to the dequeue module (XCPPDQ) to remove the request element from the request queue and return it to the freelist. The dequeue module passes control to a user abnormal-end appendage, if one exists. It similarly removes other requests related to the purged request.

- Posts the request complete, via the post routine interface (XCPPST). The completion code is X'48', as stated above. It similarly posts other requests related to the purged request.

- Passes control to another task in the system, via the I/O first-level interruption handler (location DISMISS) and the dispatcher (IEA0DS) of the supervisor.

If the channel is not busy, control is passed to the direct access Start I/O module, which sets up for the stand-alone seek operation. When this operation is accepted, the channel search module and the channel restart module continue searching for more direct access devices which need a stand-alone seek operation. When the last request element is reached, the channel search and channel restart modules start searching for requests which require data transfer operations.

If the request element contains a dummy address, there is no request element queued for the available channel. However, if the request element does not contain a dummy address, the associated UCB data transfer, busy, and not ready flags are inspected. The channel search module regains control if any of the UCB flags are on to indicate that the device is not available.

If the request has been purged and left on the logical channel queue, the sense subroutine is entered to issue a stand-alone SIO to release the device.

When a data transfer operation is started, the channel is busy for any further requests, and the channel restart subroutine again enables for any pending I/O interruptions. If an interruption occurs, the interruption supervisor receives control from the I/O FLIH. If no interruption occurs, the exit procedure tests flags indicating additional channels to be restarted (from previously handled interruptions). If one or more channels are flagged, the channel restart subroutine is re-entered. If no channels remain to be restarted, control is returned to the I/O FLIH.

CHANNEL SEARCH MODULES

The channel search modules supply the addresses of the logical channel queues for a particular channel to the channel restart subroutine until an I/O request is found that can be started on the channel or until it is determined that there is no such I/O request. There is a channel search module for each of the following:

- Multiplexor channel with burst devices attached.
- Multiplexor channel with no burst devices attached.
- Selector channel included in only one logical channel.
- Selector channel included in two or more logical channels.

Module Selection: The channel restart subroutine multiplies the address of the available physical channel by four and then adds it to the starting address of the channel table. The entry at the resultant address contains the address of the channel search module to be used for that channel.

Multiplexor Channel With Burst Devices

Whenever a burst device is operating on a multiplexor channel, that channel is tied up and other devices cannot be serviced. Because of this, those devices may overrun.

When data is transferred to or from an unbuffered control unit, overrun can occur if the channel fails to respond on time to a request for service from a device. The data (input or output) is considered invalid. Overrun also occurs during command chaining when a device receives a command that is too late for the data.

The devices are classified within the UCB as being burst, overrunable byte, or non-overrunable byte. The following rules are applied to their operation:

- Only one burst device can be operated at one time.

- A burst device and an overrunable byte device cannot operate at the same time.

- More than one overrunable byte device can operate at the same time.

- A non-overrunable byte device can operate at any time.

Burst Device: Whenever a request is for a burst device, or whenever a request element representing a request for a burst device is the first element in the logical channel queue, it must be determined whether or not any other burst devices or an overrunable byte device are operating. If they are, the request is not started and control is returned to the task supervisor. No other request elements in the logical channel queue are inspected until that burst request is started.

The request is started if no other burst device or overrunable byte device is operating. A flag byte is then set to 'FF' to indicate that a burst request is operating. (The flag byte is reset when the request is completed.) Either the channel restart subroutine is entered with the request element for the burst request or control is returned to the select subroutine.

Overrunable Byte Device: Whenever a request element for a request on an overrunable byte device is the first request in the logical channel queue, it must be determined whether or not any burst devices are operating. If none are, the channel restart subroutine is entered and the request is started. A counter is incremented every time an overrunable byte device is started and decremented every time an overrunable byte device terminates its operations.

If a burst device is operating, control is returned to the task supervisor. No other request elements are inspected until that request is started.

Non-Overrunable Byte Device: Whenever the request element for a non-overrunable byte device is the first element in the logical channel queue, the request is started regardless of the types of devices operating on the multiplexor channel.

Multiplexor Channel With No Burst Devices

The address of the logical channel queue associated with the multiplexor channel is passed to the channel restart subroutine. If the request on this queue can be started, the select subroutine is given control. Otherwise, the module is re-entered.

When the addresses of all request elements queued in the logical channel queue have been checked or when none of the I/O requests represented by the request elements could be started, control returns to the task supervisor.

Selector Channel Included in Only One Logical Channel

To start seek overlap, this module passes to the channel restart subroutine the address of the logical channel queue. The channel restart subroutine searches the request elements in this queue for a direct access device and then checks the availability of the device. If the device is busy, the subroutine searches the queue for the next request element associated with a direct access device. If the device is not busy, a seek is initiated. The channel search module is then reentered until all possible seek requests on the logical channel queue are initiated.

After all seeks have been initiated so that seek overlap is started, the same logical channel queue for the channel is searched until a data transfer request is started on the channel, or until it is determined that there are no requests that can be started. Control returns to the task supervisor.

Selector Channel Included in More Than One Logical Channel

This module starts seek overlap in the same manner as the module for a selector channel included in only one logical channel, described above. It then starts the highest priority request in each logical channel queue for the available selector channel. (When priority queuing is not implemented, the address of the first request element is passed to the channel restart subroutine.)

To determine the highest priority request elements, the module places the 2-byte link fields from the first request elements in each logical channel queue for the selector channel into the scratch field of the logical channel word for the associated queue. The scratch field maintains the address of the request element within the queue that is next to be compared for priority. The priority of the request elements are then compared.

The address of the highest priority request element is passed to the channel restart subroutine, which determines whether or not the represented request can be started. If it can, the select subroutine is entered. Otherwise, the address of the next highest priority request element is passed. This procedure continues until a request is started, or until all queues have been inspected and it is determined that no request can be started on that selector channel. Control then returns to the task supervisor.

SENSE SUBROUTINE (INTSEN)

The sense subroutine is entered when the analysis of the CSW status bits after an I/O interruption shows that the unit check bit is set. The subroutine issues a sense command to read into the UCB additional information regarding the unit check condition for the device in error.

The sense subroutine is entered from the channel restart subroutine which uses the sense subroutine as a stand-alone SIO routine to issue a release command to a shared direct access device when a request which had a stand-alone seek in progress is purged. After the release is performed, the request element is dequeued and made available for reuse.

Issuing the Sense Command: Before a sense command can be issued, the sense subroutine must inspect the UCB busy flag to determine whether or not the device represented by the UCB is busy.

If the device is busy, the sense subroutine sets the IOB sense flag to indicate that an error occurred. The I/O interruption supervisor checks the IOB sense flag at device end for the busy device and, if it is on, "ORs" the CSW field from the IOB (stored there at channel end) to the present CSW bits. Thus, the channel end error appears as a channel end/device end error and the sense command is issued.

If the device is a shared direct access device, a Read Home Address command and a Read Record Zero command are chained to the sense command to provide HA and R0 information for the error routines. The error recovery routines normally return to the I/O supervisor when this information is required to accomplish defective track recovery.

When the device is available, a CCW that contains the sense operation code is constructed and the SIO instruction is issued. The sense information is read into the UCB sense area. If the initial response to the SIO instruction is busy, the SIO is reissued. If the CSW was stored on the SIO operation, the status is tested for channel errors. If any channel errors are present, SER/CCH is entered. If unit check occurs on SIO, equipment check is indicated in the first sense byte in the UCB and the second UCB sense byte is set to X'FF' to indicate the malfunction. This status is then moved into the IOB field.

CSW Inspection: The sense subroutine "spins" on a TIO instruction until the CSW is stored, unless the device is a shared direct access device or a teleprocessing device. In these cases, the UCB status modifier flag is set to route the interruption back to the sense routine and control is returned to the I/O interruption handler. If the sense subroutine does wait for the CSW to be stored, it performs the following procedures for resultant CSW conditions:

- Channel error. The SER or CCH interface is entered.
- Unit check. Equipment check is set in the first sense byte of the UCB. X'FF' is set in the second UCB sense byte. This status is moved to the IOB sense field.
- CSW busy bit. The SIO instruction is re-issued.
- CSW status modifier bit. The UCB status modifier flag is set and control returns to the I/O interruption supervisor.

All other CSW conditions are disregarded, and the first two bytes of sense information is stored in the IOB. Control is returned to the I/O interruption supervisor.

INTERFACES

Interfaces are routines that are used as common points of departure from, and that direct the transfer of control from, an operating system routine to another system routine. The interface positions data in the form required by the system routine to be entered and also provides a return point. The following interfaces are used by the I/O supervisor (major entry points are shown within parentheses):

- Attention routine interface (INATT)
- Error routine interface (INTERR)
- SER interface (SERR04)
- ABTERM routine interface (XABEOT)
- Post routine interface (XCPPST)

Attention Routine Interface

The attention routine interface routes control to the proper attention routine after an attention interruption occurs. (See "Introduction: Attention Table.")

The appropriate attention routine is indicated by the index value to the attention table that is contained in the ATNTAB field of the UCB that represents the device requiring the attention routine. The user sets the index value when execution of an attention routine that was specified at SYSGEN time is desired. The attention routine interface adds that index value to the starting address of the attention table. The entry at the resultant address contains either the address of the pertinent attention routine or, if the index value is zero, a branch back to the I/O interruption supervisor.

After the attention routine has been executed, control returns to the I/O supervisor.

Error Routine Interface (INTERR)

The error routine interface controls execution of the error routines required by the I/O supervisor for error recovery procedures. It tests the I/O supervisor error key (contained in the DCB) and determines whether or not error recovery procedures are to be used. If a transient error routine must be scheduled, the error routine interface routes control to the exit effector. In the case of a related request with a permanent error, the error routine interface calls the I/O purge routine, so that the related request elements for the data set will be purged.

I/O Supervisor Error Key Test: Providing that the IOB does not indicate a permanent error, in which case the post routine interface is entered, the I/O supervisor error key is tested to determine when IBM-supplied error routines are to be used. They may be used for:

- All errors. For direct access devices, the error routine interface branches to the resident portion of the direct access device error routine, which handles the error or calls in transient routines. For sequential access devices, the error routine interface passes the request element to the exit effector for subsequent scheduling of the appropriate transient IBM-supplied error routine.

- No errors. The error is posted as permanent.

Scheduling of an Asynchronous Routine: If the I/O supervisor error key indicates that an IBM-supplied error routine is to handle an error, the error routine interface branches to the direct access device error routine or, if the error routine is transient, causes it to be scheduled asynchronously. Chart 14 shows the scheduling procedure.

For direct access devices, the resident portion of the direct access device error routine is entered, and if necessary, the transient portions are called in.

For sequential access devices, the error routine interface passes the address of the request element in error to the exit effector of the task supervisor. The exit effector places this element into the asynchronous exit queue for scheduling. The exit effector then completes the name of the required error routine by use of the ERRTAB byte in the UCB. This byte, when appended to the name in the system interrupt request block (SIRB) yields the name of the error routine within the SVC library. (If the freelist is empty, the error routine interface returns the request element in error to the next available, and enters the ABTERM routine interface. The associated task is abnormally terminated because in scheduling the required error routine, the contents supervisor uses the EXCP macro instruction.)

The BLDL routine of data management (IECPBLDL) uses the 8-character name to search the SVC library for the error routine. When found, the error routine is read into the I/O supervisor transient area, and then given control.

The I/O purge routine is called into the I/O supervisor transient area in the same manner as the transient IBM-supplied error routines. The error routine interface passes the request element for the related I/O request to the exit effector, which places it into the asynchronous exit queue for scheduling of the routine.

The constant that is used to complete the name of an error routine to be scheduled asynchronously, which is normally obtained from the ERRTAB byte in the UCB, resides however, for the I/O purge routine, in a location following the UCB lookup table. The address of a location that reflects this change is placed into the UCB address field of the request element. Thus, this location simulates that of the ERRTAB byte of the UCB, and the exit effector uses it to complete the name of the I/O purge routine within the SVC library. When the I/O purge routine gains control, it replaces the UCB address field in the request element with the actual UCB address for the device so that later processing is not affected.

I/O Purge Routine: The I/O purge routine is transient and is called by the error routine interface whenever a related I/O request develops a permanent error condition. (The name of the I/O purge load module is IEC0025E.) All other related I/O requests for the same data set must be purged because they are dependent upon the successful completion of the preceding related I/O requests, one of which has failed.

The I/O purge routine removes all queued request elements for the related I/O requests from the logical channel queues that are associated with the particular data set extent(s) defined in the DEB. (The correct queues are found by means of the logical channel word index indicated in the UCB.) The I/O purge routine returns the request elements to the freelist, chains the IOBs associated with them to the DEB system purge chain field, and posts the ECB that corresponds to each IOB complete with a permanent error code.

The I/O purge routine returns control to the I/O supervisor via SVC 15 and the abnormal end appendage exit is taken.

SER or CCH Interface

The SER or CCH interface (SERR04)[1] is entered whenever the I/O supervisor detects a catastrophic error (i.e., when a CSW channel data check, channel control check, or interface control check bit is set). It routes control one of the following recovery management routines:

- SER0 or SER1.
- The channel check handler.
- The machine check handler.

The choice depends on the option selected at system generation.

If the channel check handler is present, the CCH interface branches to it. (See "Channel Check Handler.")

If the channel check handler is not present, the SER interface places the address of the request element in error into location 280 and places the unit address at location 58 of the I/O old PSW. The hexadecimal code 0F, which indicates an I/O channel failure, is then placed into location 115 of the machine check (MCH) new PSW. The MCH new PSW indicates the wait state and has all interruptions masked as disabled. The SER interface then loads the MCH new PSW. The loading of the MCH new PSW gives control to either a SER routine or the machine check handler. Hardware indicators can then be inspected.

ABTERM Routine Interface

The ABTERM routine interface routes control to the ABTERM service routine of the task supervisor, which abnormally terminates a task.

The ABTERM routine interface is entered from the I/O supervisor whenever the following occur:

- Invalid DEB or UCB identification field.

- DCB specification error in the DEB.

- The protection key specified in the DEB does not match the protection key specified in the TCB.

- (MVT only) The protection key in the request element is nonzero and is not the same as the storage protection key of the IOB, DCB, and ECB. The protection key comparison is made at interruption time.

The I/O request associated with these conditions is not posted complete.

Post Routine Interface

The post routine interface routes control to the post routine of the task supervisor. The post routine interface prepares the 30-bit completion code that describes the manner in which an I/O request has completed and passes it to the post routine (IEAOPT01), where the code is posted in the ECB for inspection by the user. It also passes the address of the TCB of the task for which the I/O request has completed.

If a WAIT macro instruction had been issued, the post routine clears the wait bit in the ECB. The post routine also clears the wait bit in the request block.

The post routine interface is also used by the purge complete subroutine to post the completion of an SVC purge request (described in Section IV).

CHANNEL-CHECK HANDLER (CATNIP)

The channel-check handler (CCH) is an optional extension of the I/O supervisor on configurations using the 2860/2870 channels. (If the system is generated to operate in the multiprocessing mode, CCH is required as a recovery function.) It facilitates recovery from channel-check conditions (channel control check and interface control check), so that these conditions are not necessarily catastrophic.

The two major functions performed by CCH are the formation of the error recovery procedure interface bytes (the ERPIB) and the formation of the record entry. The interface bytes provide information to the IBM-supplied device dependent error routines which attempt to set up a retry of the failing operation. When the device dependent error routine is subsequently entered, if there are no interface bytes for the error, the device dependent error routine assumes a channel error to be permanent. The flow of the channel-check handler is shown in Chart 13.

On detection of a channel-check, the CCH can be entered from seven points within the I/O supervisor. The four types of failures handled by CCH and the points within the I/O supervisor from which entry can be made are:

----

[1] The CCH interface is part of the I/O Supervisor only if the channel check handler has been specified at system generation.

Sense Subroutine
    Two points in the sense subroutine: following an SIO or TIO instruction.

Stand Alone Seek
    The post-start I/O subroutine.

    Two points in the direct access start I/O module: following a stand alone seek SIO instruction or a seek TIO instruction.

I/O Interruption Supervisor failure
    One point within the I/O interruption supervisor: upon detection of a channel check on initial entry to the routine.

Halt I/O failure
    The Halt I/O subroutine.

Before branching to the CCH, the routine detecting the channel check identifies itself by placing a unique code in a predefined register.

Upon entry, the CCH places a code in the machine-check new PSW to identify itself to the Machine-Check Handler (MCH) program should a machine check occur during CCH processing. CCH then locates available areas for forming the ERPIB and the record entry. The interface bytes will contain the results of CCH analysis for use by the device dependent error routine, and the record entry will contain a description of the environment of the channel failure. The record entry is formed by CCH and is later written by the statistical data recorder, on the SYS1.LOGREC data set.

The areas for building the interface bytes and record entry are located by way of the communication scheme illustrated in Figure 6. The addresses are established at system generation, and the flag field is initially set to zero.

If one or more of the following conditions exists, normal CCH processing continues:

- The UCB post and/or busy bit is set.
- The error occurred in the sense routine.
- The CSW attention or device end bit is set.

If none of these conditions exists, CCH exits immediately by loading the machine check new PSW.

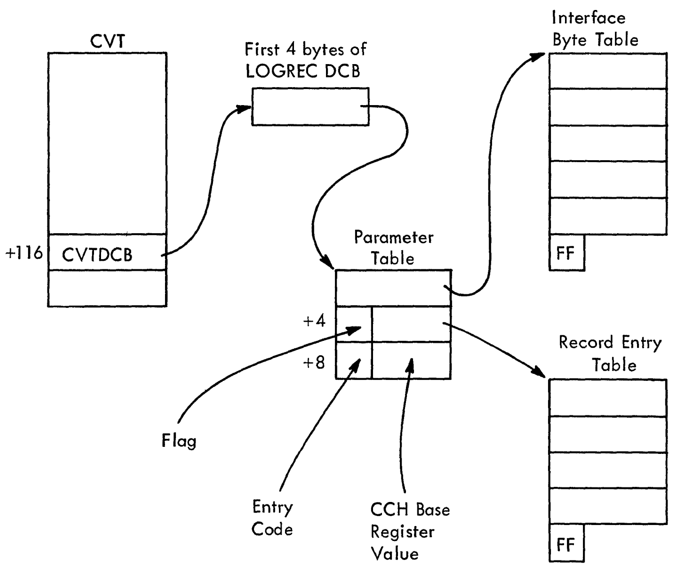


Figure 6. Channel-Check Handler Communication Scheme

If the failure occurred on a multiplexor channel, CCH determines whether a system reset has occurred. If a system reset has occurred, CCH , after performing its analysis, places the system in the wait state.

The record entry is always produced unless space for the record entry is not available. This can occur when a series of channel errors has occurred without opportunity for the statistical data recorder to write them on the SYS1.LOGREC data set. The ERPIB are produced unless the channel check occurred in the sense subroutine and was corrected or unless it followed a Halt I/O instruction. If the channel check occurred in the sense subroutine, control may be returned to the sense subroutine for retry of the sense operation. However, if a channel check is found not suitable for a retry in the CCH sense intercept routine, or if after three attempts the sense operation is not successfully completed, the ERPIB are produced with the 'no retry' bit set.

In the case of a Halt I/O instruction, retry is not necessary because the effect of the channel check is the same as that of the instruction. Control can be returned to the instruction following the Halt I/O.

As a result of the CCH operation, execution can continue in the I/O supervisor with the scheduling of a device dependent error recovery procedure to retry the failing operation, as in the case of a non-catastrophic error. The last function performed by CCH before relinquishing control is to remove its identifying code from the machine-check new PSW.

Section III: IBM-Supplied Error Routines

A device-dependent error routine may be entered following a unit check, a unit exception, a wrong length indication, a program check, a protection check, a chaining check or a channel data check indication in the CSW status bytes. In addition, if the optional channel-check handler (CCH) is included at system generation time, a device-dependent error routine is entered following a channel control check or an interface control check. When such a condition exists, the I/O supervisor passes control to the error routine interface, which tests the I/O supervisor error key in the DCB, which indicates whether or not an error routine is to be used, and if one is, branches to or has the appropriate error routine scheduled asynchronously. Chart 14 shows the scheduling of an IBM-supplied error routine.

There are three types of IBM-supplied error routines:

- Device-dependent routines
- Common routines
- I/O inboard and outboard recording routines

Device-dependent routines attempt error recovery and correct errors for particular device types according to the standard IBM error recovery procedures (ERP). All of the device-dependent routines, except a resident portion of the error routine for direct access devices, use the I/O supervisor transient area for execution.

Common routines are used by the device-dependent routines. The common routines are the error interpreter, which is resident, and the write-to-operator and statistics update routines, which are transient. Also, the I/O inboard and outboard recording routines are used as an extension of the device-dependent error routines.

The I/O inboard routine writes on the system residence device channel inboard records that have been created and formatted by the channel-check handler (CCH). I/O outboard recording routines maintain external records on the system residence device. These records contain information regarding the operation of each I/O device. These routines are the statistical data recorder and the outboard recorder; they are always provided when IBM-supplied error routines are included in the system.

The device-dependent and common routines determine the type of error. When the error is such that the I/O request can be retried, the error routine issues an SVC 15 to retry the channel program, and then returns to the task supervisor by means of SVC 3. As each retry is executed, control returns to the error routine.

When an error is corrected by an IBM-supplied error routine, the routine issues SVC 15 and an SVC 3 to return control to the I/O supervisor. The I/O supervisor posts the ECB with a normal completion code and normal processing continues.

At the completion of error processing (successful or unsuccessful), the common error routines, when necessary, update the statistics table, write error messages to the operator, and call the I/O outboard recording routines for external error recording.

Also, when the IBM-supplied error routine cannot correct the error, the routine issues SVC 15. Upon receiving control the I/O supervisor purges any request elements related to the request element in error by use of the I/O purge routine. The I/O supervisor points to the purged requests with the system purge field in the DEB for the appropriate data set. The purge routine returns the related request elements to the freelist and posts the ECB with a purge code.

Device-Dependent Routines

The device-dependent routines correct errors according to the standard IBM error recovery procedures. The routines are:

- 1052 (2150) error routine, which handles errors on the console typewriter. (The load module name is IGE0000D.)
- 2540 (2821) error routine, which handles card read and punch errors. (The load module names are IGE0001C and IGE0101C.)
- 1403, 1443 error routine, which handles printer errors. (The load module name is IGE0000G.)
- 1442, 2501, 2520 error routine, which handles card read and punch errors. (The load module name is IGE0000E.)

- 2671 (2822) paper tape reader error routine, which handles errors on the paper tape reader. (The load module name is IGE0002I.)

- 2400 tape series (including the 2420) error routine, which handles magnetic tape errors. (The load module names are IGE0000I, IGE0100I, IGE0200I, and IGE0900I.)

- Direct access device error routines, which handle errors on direct access devices and perform alternate track procedures. (The load module name for the transient portion is IGE0000A.) Most of this error routine resides in main storage; the load module for the resident portion can be one of the following, depending upon device type and selection of the track overflow feature:

  IEC23XXB - 2311 without overflow

  IEC23XXC - 2311 with overflow

  IEC23XXD - 2311 and/or 2302, 2303, without overflow

  IEC23XXE - 2311 and/or 2301, 2302, 2303, 2314, with overflow

  Note 1: On the 2301 and 2314 the overflow is a standard feature.

  Note 2: In all of the above modules, code is included for the 2321 to record statistics. All 2321 error handling, however, is performed in transient modules.

- 2321 data cell device error routine, which handles errors on the 2321 data cell drive. The 2321 data cell device error routine is composed of five transient load modules: IGE0002H, IGE0102H, IGE0202H, IGE0302H, and IGE0402H. When an error is detected for a direct access device, one of the above resident direct access routines is entered. If the resident direct access routine detects a 2321 error, control is passed to the first 2321 transient load module (IGE0002H).

- 2250 error routine, which handles errors on the 2250 display unit. (The load module name is IGE0010A.)

- 2260 error routine, which handles errors on the 2260 display station and associated 1053 printer. (The load module names are IGE0010B and IGE0110B.)

- 2280/82 error routine, which handles errors on the 2280 film recorder and the 2282 film recorder/scanner. (The load module name is IGE0010C.)

GENERAL OPERATING PROCEDURE

The device-dependent routines, with the exception of the resident direct access device routine, gain control through the contents supervisor in PCP (or via the stage-3 exit effector in MVT). Although each routine has characteristics pertinent to the device type, the programming techniques used for each routine are similar.

The routines set the IOB error indicator bits, examine the sense and CSW status bits stored in the IOB to determine the type of error, and where possible, attempt to recover the error by retrying the channel program by means of SVC 15.

Chart 15 shows the general operating procedure for all device-dependent routines except the 2250, 2260, and 2280/82 error routines. The general operating procedure for these routines is shown in Chart 16.

Setting Error Indicators: All device-dependent routines upon entry, turn on the IOB exception bit and error flag. These flags indicate to the error routine interface that an error routine is in control, and that control must be returned to the error routine until the error is corrected or is determined to be uncorrectable. An uncorrectable error is a permanent error.

When the error is permanent, the IOB exception bit is left on and all error flags and counts are reset to zero. If the DCB exception bits are set to 01 (i.e., indicate that error correction is in progress), the I/O supervisor resets the bits to 11, indicating the permanent error condition. Until the DCB exception of 11 is cleared, all further related requests for the particular DEB are rejected and posted as a permanent error.

When the error is corrected, the IOB and DCB exception flags and the IOB error flags and counts are reset to zero.

IOB Channel Program Description Bit Test: In all error routines, except the direct access device, 2671/2822, 1052, 2150, 2250, 2260, and 2280/82 error routines, the IOB channel program description bits are inspected for the type of chaining within a channel program in error so that the proper restart procedure is taken. The direct access device, 2671/2822, 1052, 2150, 2250, 2260, and 2280/82 error routines always retry the channel program in error from the top of the CCW list (at the first CCW);

thus, they handle any type of channel program and chaining is not significant.

The procedure taken for chained channel programs, shown in Chart 15, is as follows:

- Data and Command Chaining. The error routines do not retry any channel programs that have mixed chaining (IOB channel program description bits set to 11); the routines write an error message to the operator and indicate a permanent error by means of the DCB exception bits. SVC 15 is issued so that the request element will be returned to the freelist, and then control is returned to the task supervisor via an SVC 3 instruction.

- Command Chaining. When a channel program is command chained (IOB channel program description bits set to 01), it must be restarted at the proper CCW address. When a channel program has been retried and the error is corrected, a subsequent error may occuron a different CCW. This error is detected when the address of the erroneous CCW does not equal the restart address in the IOB. In this case, the channel program is restarted at the erroneous CCW address. However, if an error occurs at the execution of the SIO instruction for this retry, the accuracy of the CCW is unpredictable. This error is indicated by the condition code setting of 01. The channel program is again started at the address of the erroneous CCW, which is the restart address in the IOB.

- Data Chaining or No Chaining. When the channel program has data chaining or no chaining (IOB channel program description bits set to 10 or 00, respectively), the channel program is restarted from the top of the CCW list, which is the start address in the IOB.

Determining the Type of Error: The type of error is indicated by the setting of the CSW status bits, and in the case of a unit check condition, the sense bits. The error routines (except the 2250, 2260, and 2280/82 error routines) examine the setting of these bits by means of the error interpreter routine. Instructions within the 2250, 2260, and 2280/82 error routines test the bits; the other device-dependent routines branch to the error interpreter with a list of codes and address constants following the branch instruction.

Each entry in the list is 2 bytes in length. The order of the entries depends upon IBM standard error recovery procedures, upon the type of device-dependent routine in control, and upon the meaning the CSW status or sense bits have for the type of device.

Each entry contains a 1-byte code that corresponds to the relative position of either a CSW status bit or a sense bit and a 1-byte index factor that indicates the point at which the error interpreter is to return to the error routine after determining the type of error.

The error interpreter routine compares each code to the CSW status or sense bit. If the particular status or sense bit is set, the error routine regains control at the point indicated by the address constant.

The error routine then handles the condition denoted by the particular status or sense bit. The following procedures are taken:

- Permanent error. A permanent error is one that either cannot be corrected or fails to be corrected after standard error recovery procedures. A permanent error is indicated by the IOB exception bit. The statistics update and the I/O outboard recording routines are called for error recording and then SVC 15 is issued.

  After the I/O supervisor routines have posted the request complete with a permanent error, have entered the ABTERM routine interface in the case of a program or protection check, have purged any requests related to the request in error, and have entered the abnormal end appendage, control returns to the error routine. The error routine uses SVC 3 to return to the task supervisor.

- Operator Intervention. When it is determined that operator intervention is required, the write-to-operator routine is entered for the writing of the "intervention required" message. SVC 15 is issued, then SVC 3 is issued to return control to the task supervisor.

- Recoverable errors. The channel program is retried by means of the SVC 15 instruction, and if the error is corrected, the IOB exception and error flags are set to zero and normal processing continues.

  If the error is uncorrectable, the procedure for a permanent error is taken.

Statistics Update: In unit check cases, the statistics update routine is always called to update the error counters in the statistics table and, if a statistics counter overflows, the statistical data recorder is called for external error recording.

DEVICE-DEPENDENT CHARACTERISTICS

Since each device type has different characteristics, the meaning of the CSW unit exception bit, the sense bits, IOB flags 1, flags 2, and flags 3 fields, and the IOB error count field vary according to the type of device. The CSW status bit interpretation and their order of inspection is shown in the coding for the IBM-supplied error routines. The sense bit meanings are shown in Table 2 and are explained in the appropriate component description publication for the device. The IOB flags 1, 2, and 3 fields and the IOB error count field are also shown in Table 2.

1052 (2150) Error Routine (IGE0000D)

Special considerations are necessary for errors on the console typewriter. Messages can be issued through the common write-to-operator routine only for read errors. The operator is notified of other types of errors by use of the console alarm bell.

The following CSW conditions are treated as permanent errors: program check, unit exception, protection check, channel data check, channel control check, interface control check, chaining check, and incorrect length. Control is returned immediately from the 1052 error routine to error EXCP for abnormal termination of the task.

The 1052 error routine tests bits M0 and M1 in the third byte of the IOB flags to control its procedure on subsequent entries after the original error. If both are set to 0, the common interpreter is entered to determine the type of error. If M0 has been set to 1, it indicates error EXCP was entered to ring the console alarm to signal a permanent error. Control is returned via SVC 15 for abnormal termination. If M1 has been set on, intervention required has been signalled, and the original channel program is to be retried.

Actions taken to recover from error conditions indicated in sense byte 0 are described below.

Bit 0: Command Reject.

This bit is set when an invalid command code is detected. The IOB error flag is turned off and the IOB exception flag turned on, indicating a permanent error. Control is returned to I/O supervisor via SVC 15 for abnormal termination.

Bit 1: Intervention Required.

This bit is set when the printer runs out of forms or when the not-ready key has been depressed. The error routine places the control command to ring the console alarm in the reposition modifier field of the IOB. Modifier flag 1 is set in the IOB to signal error EXCP to build a CCW employing this command code. Bit M1 is set in the IOB and control returned by SVC 15. The intervention required condition does not terminate a read or write operation that is already in progress.

Bit 2: Bus-Out Check.

This bit is set when a parity error is detected on a command or data byte. The 1052 error routine sets the bus out flag in the IOB and retries the user's channel program via SVC 15. If the error recurs, the control command to ring the console alarm is placed in the reposition modifier field of the IOB, and modifier flag 1 is set to signal error EXCP to construct a CCW with this command. IOB bit M0 is set, and control returned via SVC 15.

Bit 3: Equipment Check

This bit is set when one of the following occurs:

- Keyboard parity error.
- Keyboard-printer compare check.
- Printer failed to take a mechanical cycle during a read or write operation.

If the error occurred on a read command, a permanent error is assumed and the write-to-operator routine is called; it writes a message indicating failure to read input messages. For write errors, the error routine retries the channel program via SVC 15. On the second occurrence of the error, the console bell is rung via the procedure described above for bus-out check.

Bits 4-7: Should not occur.

If these bits are set, a catastrophic error is assumed. A permanent error is signalled to error EXCP and abnormal termination takes place.

Table 2. Sense Bytes, IOB Flags, and IOB Error Counts

Bit Byte	0	1	2	3	4	5	6	7
1052 (2150)								
Sense 0	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	———	———	———	———
IOBFLG1	Channel Program Description		IOB Error Flag	Mode 1 Flag	———	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG3	———	———	Write Error Count	Bus-Out Error Count	Control Bit M1	Message Type	Control Bit M0	Logout Flag
2540 (2821)								
Sense 0	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	Data Check	———	Unusual Command Sequence	———
IOBFLG1	Channel Program Description		IOB Error Flag	———	———	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG2	———	———	———	———	———	———	———	QSAM Access Method
IOBFLG3	Indicator 1	Entry Flag	Read Error Count	Bus-Out Error Count	Punch Relay	Message Type	QSAM Count	Logout Flag
IOB Error Counts Byte 1	———	———	———	———	———	———	———	Data Check Count
1403 , 1443								
Sense 0	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	Data Check / Type Bar	Parity Error / Type Bar	———	Channel 9
IOBFLG1	Channel Program Description		IOB Error Flag	———	———	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG2	———	Sense Bit	Purge Flag	———	———	———	———	———
IOBFLG3	———	IOB Entry Bit	Load Gen in UCS Parity Error Count	Bus-Out Error Count	———	Message Type		Logout Flag
1442 , 2501 , 2520								
Sense 0	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	Data Check	Overrun	———	———
IOBFLG1	Channel Program Description		IOB Error Flag	———	———	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG2	———	Sense Bit	Purge Flag	———	———	———	———	———
IOBFLG3	———	IOB Entry Bit	Read Error Count	Bus-Out Error Count	Data Check Flag	Message Type	Overrun Count	Logout Flag
2671 (2822)								
Sense 0	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	Data Check	———	———	———
IOBFLG1	Channel Program Description		IOB Error Flag	———	———	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG2	———	Sense Bit	Purge Flag	———	———	———	———	———
IOBFLG3	———	Equipment Check Count		Bus-Out Error Count	———	Message Type	———	Logout Flag
2400								
Sense 0	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	Data Check	Overrun	Word Count Zero	Data Converter Check
Sense 1	Noise	Unit Status A	Unit Status B	7-Track Tape Unit	Tape At Load Point	Unit In Write Status	File Protect Status	Not Capable
Sense 2	———	———	———	———	———	———	Multi Error	Track Exists
Sense 3	R/W VRC	LRCR	Skew	CRC	Skew Reg VRC	Phase Encoded	Backward Status	C-Compare
Sense 4	———	Reject TU	Read Clock	Write Clock	Delay Counter	C Seq	B Seq	A Seq
IOBFLG1	Channel Program Description		IOB Error Flag	Tape Reposition	CRC Bit	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG2	———	Sense Bit	Purge Flag	———	Read/Write Unit Exception	Use UCB RORCCW	———	———
IOBFLG3	Noise Message Given	IOB Entry Bit	Tape Cleaning Bit	Control Flag	Read Opposite Recovery	Message Type	Control Unit Busy	Logout
IOB Error Counts (Byte 1)	ICC or CCC	Overrun, Bus-Out, Channel Data Check, Chaining Check			———	Cleaner Control Count		
IOB Error Counts (Byte 2)	Write Indicator	Read Data Check Counts ERP ERG Data Check			User ERG Data Check Write Check Counts			
Bit Byte	0	1	2	3	4	5	6	7

Bit Byte	0	1	2	3	4	5	6	7
2301 (2820) ; 2302 , 2303 , 2311 , 2314 , 2321 , (2841)								
Sense 0 2301 (2820) Only	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	Data Check	Overrun	———	Invalid Address
Sense 0 All Other DA Devices	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	Data Check	Overrun	Track Condition Check	Seek Check
Sense 1 2301 (2820) Only	Data Check In Count Area	Track Overrun	End Of Cylinder	Invalid Command Sequence	No Record Found	File Protect	Service Overrun	Overflow Incomplete
Sense 1 All Other DA Devices	Data Check In Count Area	Track Overrun	End Of Cylinder	Invalid Command Sequence	No Record Found	File Protect	Missing Address Marker	Overflow Incomplete
IOBFLG1	Channel Program Description		IOB Error Flag	———	———	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG2	———	Sense Bit	Purge Flag	Read HA Flag	No End Extent Flag	CCHH Update Flag	———	———
IOBFLG3 2321 (2841) Only	Indicator 1	Track Condition Flag	No Record Found	Pick Flag	Restore Flag	Message Type	Sweep Flag	Logout Flag
IOBFLG3 All Other DA Devices	Indicator 1	Track Condition Flag	No Record Found	Bus-Out Error Count	Restore Flag	Message Type	———	Logout Flag
IOB Error Cnt, Byte 1 2321 Only	Bits Incremented for: No Record Fnd; Chaining, Bus-Out, Seek, Data Checks; Overrun And Missing Address Marker				Bits Incremented For Either Data Check Or Missing Address Marker			
IOB Error Cnt, Byte 2 2321 Only	This Byte Incremented For Data Check Or Missing Address Marker							
IOB Error Cnt, Byte 1 2311, 2301	Data Check Count							
IOB Error Cnt, Byte 2 2311, 2301	Write Indicator	Overrun Count Or Chaining			No Record Found, Missing Addr Marker, Seek Check			
2250								
Sense 0	Command Reject	———	Bus-Out Check	Equipment Check (Model 2)	Data Check	———	Buffer Running	———
Sense 1	Light Pen Detect	End Order Sequence	Character Mode	———	———	2840 Output Check	2840 Input Check	———
IOBFLG1	Channel Program Description		IOB Error Flag	———	———	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG2	Halt I/O (HIO) Issued	Sense Flag	———	———	———	———	Film Unit Flag	———
IOBFLG3	———	———	———	———	———	Message Type	———	Logout Flag
IOB Error Counts (Byte 1)	Bus-Out Count 1	Bus-Out Count 2	Data Check	Equipment Check	2840 Input Check	2840 Output Check	AB. End Appendage Bit	———
2260 (1053)								
Sense 0	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	———	———	———	———
IOBFLG1	Channel Program Description		IOB Error Flag	———	———	IOB Exception Flag	———	Start/ Restart Flag
IOBFLG2	Halt I/O (HIO) Issued	Sense Flag	———	———	———	———	Film Unit Flag	———
IOBFLG3	———	———	———	———	———	Message Type	———	Logout Flag
IOB Error Counts (Byte 1)	Bus-Out Count 1	Bus-Out Count 2	Bus-Out Count 3	Error Rtn Channel Program	———	———	AB. End Appendage Bit	Check
2280 (2282)								
Sense 0	Command Reject	Inter-vention Required	Bus-Out Check	Equipment Check	Data Check	———	———	Illegal Sequence
Sense 1	Read Count Check	Recorder Film Low	Recorder Forced Gap	Film Motion Limit (2282)	———	2840 Output Check	2840 Input Check	Graphic Check
IOBFLG1	Channel Program Description		IOB Error Flag	———	———	IOB Exception Flag	I/O Related Flag	Start/ Restart Flag
IOBFLG2	Halt I/O (HIO) Issued	Sense Flag	———	———	———	———	Device End Post Code	———
IOBFLG3	———	———	———	———	———	Message Type	———	Logout Flag
IOB Error Counts (Byte 1)	Bus-Out Count 1	Bus-Out Count 2	Data Check	———	2840 Input Check	———	AB. End Appendage Bit	———
Bit Byte	0	1	2	3	4	5	6	7

2540 (2821) Error Routine

When a card jam causes a punch error indication on the 2540 punch, operator restart procedures are necessary.

No recovery is attemped under program control on a punch error (equipment check) unless the queued sequential access method (QSAM) is in use and requests recovery.

The 2540/2821 error routine attempts recovery from channel data checks. It treats all other channel errors as permanent.

The bits in the first sense byte (byte 0) have the following meanings for the 2540 error routine:

Bit 0: Command Reject

The setting of this bit indicates one of the following permanent errors:

- A read backward or write command was issued to the reader.
- A read backward control other than NOP or a read feed and stacker select command was issued to the punch without the PFR feature.
- A read backward or control command was issued to the punch with the PFR feature.
- An incorrect sequence of instructions was issued.
- A command was received which the device was not designed to execute.

Bit 1: Intervention Required

The setting of this bit indicates a not-ready condition due to one of the following:

- Cards are not at each station (not EOF for the Reader).
- Stacker is full.
- Hopper is empty (not EOF for the Reader).
- Stop key is depressed.
- Chipbox is full or removed.
- Card is jammed.

A message is typed, and the operator must perform the right action. Upon correction, the last operation which was interrupted is repeated.

Bit 2: Bus-Out Check

The setting of this bit indicates a parity error on bus-out on a command or data byte during either initial selection or command execution.

When this error is detected, no punching occurs. The operation is retried once and if it occurs a second time, it is considered a permanent error and a message is written.

Bit 3: Equipment Check

The setting of this bit occurs after one of the following:

- Hole count error
- Buffer parity error
- Translate check
- Address check

This error applies to the previous card punched for which data was transmitted; it is considered permanent.

Bit 4: Data Check

The setting of this bit indicates an invalid card code was detected. The operation is retried five times. On the sixth occurrence, it is denoted as a permanent error.

Bit 5: Unused

Bit 6: Unusual Command Sequence

This bit is set due to a read following a read with no intervening feed. This error is considered permanent.

Bit 7: Unused.

1403, 1443 Error Routine

The 1403/1443 error routine treats all channel errors as permanent.

The bits in the first sense byte (byte 0) have the meanings described below.

Bit 0: Command Reject

This bit is set when one of the following occurs:

- A read backward command is received.
- A carriage instruction to space more than three lines is received.
- A carriage instruction to skip to channel 0, 13, 14, or 15 is received.

- A command is received that the device has not been designed to execute.

These errors are permanent.

Bit 1: Intervention Required

This bit indicates a not-ready condition due to one of the following:

- Printer has run out of forms or forms have jammed.
- Stop key is depressed.
- Cover interlock switch is open.
- Typebar remove switch is set to the off or remove position.

A message is typed and the operator must correct the condition. When the interruption occurs due to the correction of the error, i.e., the unit goes from not-ready to ready status, the last operation is repeated.

Bit 2: Bus-Out Check

The setting of this bit indicates that a parity error has occurred during the initial selection of a control unit/ device. This error is considered permanent.

Bit 3: Equipment Check

The setting of this bit indicates a program resettable malfunction was detected in the printer or in its controls. The errors are buffer or hammer checks, and are reset by the next write or control command.

Bits 4 and 5: Type Bar Selection (1443 only)

These bits are used in combination and indicate the following:

- 52 character set (00)
- 13 character set (01)
- 39 character set (10)
- 63 character set (11)

These bits can be changed only by repositioning the type bar character indication switch.

Bit 4: Data Check (1403 only)

If the universal character set feature is installed, the setting of this bit indicates that a code in data storage did not match any code generator storage. This is considered a permanent error.

If the universal character set feature is not installed, this bit should not be set.

Bit 5: Code Generator Storage Parity Error (1403 only)

If the universal character set feature is installed, this bit indicates that a parity error was detected on data being written into or read from code generator storage. If it occurs while loading generator storage, a retry is done. If it occurs at any other time, it is treated as a permanent error.

If the universal character set feature is not installed, this bit should not be set.

Bit 6: Should Not Occur.

If this bit is set, a catastrophic error is assumed. The Log Out bit in the IOB is set and the permanent error procedure is performed.

Bit 7: Channel 9

This bit is set when a hole is sensed in channel 9 of the carriage control tape during execution of the last write or control command.

1442, 2501, 2520 Error Routine

The bits in the first sense byte (byte 0) have the meanings described below.

Bit 0: Command Reject

This bit is set when a read backward command is issued or a command is received which the device has not been designed to execute. The error is considered permanent.

Bit 1: Intervention Required

This bit is set when one of the following occurs:

- Power is off.
- Cards are not registered at the read station.
- Stacker is full.
- Feed check light is on.
- Stop key is depressed.
- Chip box is full or removed.
- No cards are in hopper and end of file is not on.
- Cover interlock is open.

A message is typed, and the operator must correct the condition. Upon correction, the last operation that was interrupted is repeated.

Bit 2: Bus-Out Check

This bit is set when there is a parity check on bus-out during command or data bytes, but not during the time period of address-out or command-out proceed.

On a command byte, one retry is made; if upon return the error is not corrected, it is considered permanent.

On a data byte, the card is already punched; the error is considered permanent.

Bit 3: Equipment Check

This bit is set at the following situations:

- Punch operations - the punch echo is checked for invalid card codes; i.e., rows 1 through 7 have more than one punch in one column.

- Read or punch operations - equipment check detects an error from the 9 bit match. The invalid card code part of equipment check is not set during card image punch.

Bit 4: Data Check

The setting of this bit indicates an invalid card code was detected during a read operation; i.e., rows 1 through 7 have more than one punch in any one column. (This is not indicated when the read command is in data mode 2.) This error pertains to the card for which data was sent.

A message is typed, and the operator must reposition the cards for one retry. If the error persists, it is considered permanent.

Bit 5: Overrun

This bit is set during a read operation if the interface has not transmitted the last previous character by the time the next character is to be read into the register.

The operator must reposition the cards and the operation is retried once. If the error persists, it is considered permanent.

Bits 6 and 7: Should Not Occur.

If either or both of these bits are set, a catastrophic error is assumed. The Log Out bit in the IOB is set and the permanent error procedure is performed.

2671 (2822) Error Routine

The 2671/2822 error routine treats all channel errors as permanent.

Because the paper tape reader cannot be backspaced or repositioned under program control, all unit check errors except a bus-out check are considered permanent. The bus-out check is retried once, but if not corrected, is then considered permanent.

The sense bits in the first sense byte (byte 0) have the meanings described below.

Bit 0: Command Reject

This bit is set for the 2671 Paper Tape Reader when a read backward or a write command is received at the tape reader or when valid commands have invalid modifier bits.

A parity error on the command prevents these conditions from being detected.

Bit 1: Intervention Required

This bit is set when the device is "not ready". This is best described by stating the conditions for ready. The device is considered ready when:

- Power is on
- Tape is properly loaded
- Start key is pressed

The tape reader becomes not ready immediately when:

- The stop key is pressed while no read operation is in progress.
- The load key is pressed.
- The power-on and reset key is pressed.

The 2671 Paper Tape Reader becomes not ready at the completion of a read operation when:

- There is an overstep condition (the tape did not stop on one character) on a normal or abnormal tape stop.

- Tape jamming occurs in the take-up stacker (special feature). The tape transport mechanism is immediately released.

- The end of tape is detected at the reading station (with EOF light off).

- The stop key has been depressed during the read operation and a normal or abnormal completion has occurred.

As soon as the device is restored from the not-ready conditions back to a ready condition, a device end interruption is generated to inform the program of this transition.

Bit 2: Bus-Out Check

This bit is set for the 2671 Paper Tape Reader when a parity error has been detected for a command issued by System/360. A transmission error or a command byte is thus separated from a programming error as an invalid command.

Bit 3: Equipment Check

This bit indicates that a time-out has been detected in the tape reader since the last data byte was sensed at the tape read station. This time-out of approximately 1 second can be caused by:

- A failure of the photo cells or the light projection system.
- A tape drive motor failure.
- An unusual tape creeping condition.
- A length of tape without punching.
- Tape jamming in the optional stackers.

Bit 4: Data Check

This bit is set for 2671 Paper Tape Reader when an invalid character is read. Parity of the data to be read can be checked if the parity switch is set to either the odd or even position. Regardless of the tape code used, the data sent to the system is in odd parity.

Bits 5-7: Should Not Occur.

If either or both of these bits are set, a catastrophic error is assumed. The Log Out bit in the IOB is set and the permanent error procedure is performed.

2400 Tape Series Error Routine

Minimum tape record lengths are 12 bytes for a read operation and 18 bytes for a write operation. This minimum length has been established to distinguish noise records from data records. There is no check for noise unless there is a data check.

If an ending status (e.g., channel end, device end, or unit check) is indicated in the CSW, the error routine increases the block count in the associated data control block (DCBBLKCT). The block count is increased by the amount shown in the "block count increment" field (offset 1C) of the associated input/output block.

If the error condition is not corrected after the specified number of retries or if no retries are attempted, control is transferred to module IGE0025C (WTO). For all error conditions except Command Reject, Load Point, Data Converter Check, and Not Capable, the IOB logout bit is set on before control is transferred to the write to operator module. This indicates to the write to operator module that it should transfer control to the outboard recorder after a message is written to the operator.

The sense bits which cause Unit Check to be set and the methods in which they are handled are described below. The bits are discussed in the same order in which they are handled.

Byte 0, Bit 3: Equipment Check
No retries are attempted.

Byte 0, Bit 2: Bus-Out Check

1. No Device End in Unit Status: The command is reissued.
2. Device End in Unit Status: If this condition occurs during a write, the tape is repositioned and the command is reissued. If this condition occurs during another type of command, the command is reissued.

This procedure is followed until five retries have been attempted.

Byte 0, Bit 1: Intervention Required

1. No Device End in Unit Status: TU Status B (sense byte 1, bit 2) is tested. If TU Status B is off, the device is non-existent. No retries are attempted. If TU Status B is on, an operator intervention required message is provided and when the drive is made ready, the command is reissued.
2. Device End in Unit Status: If the command was a Rewind-Unload, processing continues. Otherwise the intervention required condition is ignored and the checking of sense data continues.

Byte 0, Bit 0: Command Reject
No retries are attempted.

Byte 0, Bit 5: Overrun

The tape is repositioned and the command is reissued. This procedure is followed until five retries have been attempted.

Note: A data check during overrun suppresses the overrun condition.

Byte 1, Bit 4: Load Point
No retries are attempted.

Byte 0, Bit 4: Data Check

If the operation is a read or read backwards, and if the noise bit (Byte 1, Bit 0) is on or if more than 11 bytes were read, the tape is repositioned and the read or read backwards is retried. If the noise bit is off and the block size is less than 12 bytes, another block is read and operation continues. This procedure is followed until forty retries have been attempted. Every fifth retry is preceded by a tape cleaner action. A TIE (Track-in-Error) command is issued before every reread to send sense byte 2 to the tape control unit.[1] Sense byte 2 contains track-in-error information.

If the above procedure does not recover from the error, the error routine tries to recover by reading in the opposite direction. If, however, any of the following conditions exists, the routine does not attempt opposite direction recovery:

- Data chaining is being performed.
- Data conversion mode and 7-track tape are being used.
- The original CCW count is less than the physical block size on the tape.
- "Suppress data transfer" is specified in the original CCW.

In attempting opposite-direction recovery, the error routine first prepares to read without repositioning the tape. It issues a Track-in-Error command to send to the tape control unit the sense byte 2 obtained during the last original-direction retry. The routine then issues a Read or Read Backwards command as its first retry. On subsequent retries, it repositions the tape, issues a Track-in-Error command, then a Read or Read Backwards command. Before every fifth retry, the routine causes a tape-cleaner action. The routine continues its retries until either it is successful (i.e., no unit check occurs) or it has made 40 unsuccessful retries. During this recovery procedure, no temporary error statistics are accumulated by the statistical update routine (IGE0025D).

The read-opposite CCW has the "suppress data transfer" bit set until the first successful retry.[2] If successful, the routine issues a Forward Space Block or Backward Space Block command to reposition the tape past the block being read. It also alters the read-opposite CCW so that it can transfer data. The alteration consists of clearing the "suppress data transfer" bit and placing the "exact" count and the data address into the CCW. The "exact" count equals the block size to be read.

If all 40 read-opposite recovery attempts are unsuccessful, the routine makes one final retry, this time attempting to read in the original direction. Unlike previous attempts, it does not issue a Track in Error command before the Read or Read Backward command. Note: The routine avoids the final read attempt if either a permanent bus-out check occurs when the repositioning command is issued, or an equipment check occurs.

If the operation is a write or write tapemark, the tape is repositioned, an erase gap command is issued, and the write or write tapemark command is reissued. This procedure is followed until fifteen retries have been attempted.

If the operation is an erase gap, the command is reissued. This procedure is followed until three retries have been attempted.

Channel Data Check

If this condition occurs during a read or write operation, the tape is repositioned and the command is reissued. If this condition occurs during a control command, the command is reissued. This procedure is followed until five retries have been attempted.

Byte 0, Bit 7: Data Converter Check

No retries are attempted.

[1] For further information on tape commands, refer to IBM 2400 and 2816 Model 1 Component Description.

[2] This CCW is located at offset hex. 30 in the unit control block.

Byte 1, Bit 7: Not Capable

The tape is repositioned to Load Point. No retries are attempted.

No Previous Sense Bits On

No retries are attempted.

Chaining Check

The tape is repositioned and the command is reissued. This procedure is followed until five retries have been attempted.

Interface Control Check or Channel Control Check

Control is passed to the 2400 Series Tape Error Routine if the channel-check handler (CCH) is in the system and an ICC or CCC occurs.

No retry is attempted if one of the following conditions exist: (Refer to Section V: Control Block and Table Formats for a complete discussion of the ERPIB.

ERPIB Byte 4, Bit 7 on (unconditional no-retry).

ERPIB Byte 6, Bit 3 or 4 or 5 off (retry code invalid, unit status invalid, or command address invalid).

ERPIB not produced.

ERPIB Byte 7, Bit 0 and 1 both on (system reset).

ERPIB Byte 4, Bit 2 on (TIO with no pending interrupt stored a CSW).

ERPIB Byte 4, Bit 3 on (HIO stored a CSW).

ERPIB Byte 4, Bits 0 and 1 both on or both off.

ERPIB Byte 4, Bit 5 off (if Unit Check is indicated in Unit Status).

Unit Check and Equipment Check are also present.

ICC or CCC occurred during tape ERP repositioning (Unless cc=1).

If none of the above conditions exists, retries are attempted for the following ERPIB termination code and retry combinations (Byte 7).

Termination code 10, Retry Code 010
Write command -- retry once

Termination Code 01, Retry Code 011
and
Termination Code 01, Retry Code 101

Write command -- reposition and retry once

Read command -- reposition and retry once

Other Commands (excluding set mode and transfer-in-channel) -- Operation is complete and no recovery action is required.

Direct Access Device Error Routines

There are five direct access device error routines available to the user. The one used depends on the devices in the user's system. The routines are:

1. 2311,2321/2841 resident error routine without overflow.

2. 2311,2321/2841 resident error routine with overflow.

3. 2301/2820 (2302,2303,2311,2314,2321/2841) resident error routine without overflow.

4. 2301/2820 (2302,2303,2311,2314,2321/2841) resident error routine with overflow.

5. 2321/2841 transient error routine.

After gaining control, the resident direct access device error routines inspect the UCB status byte for a unit check condition. If unit check has occurred, the last two bytes of the statistics table, the statistics table work area, are updated. Then, if a 2321 error has occurred, control is passed to the first load module of the transient 2321 error recovery routine.

All direct access device error routines attempt to restart the channel program in error from its original starting point, the first CCW. This eliminates the problem of repositioning required by disk, and allows the handling of channel programs that include command and data chaining. Any errors within errors are also handled. For example, if a data check occurs and upon retry a seek check is encountered, the error counts for both the data check and seek check are incremented and recovery is still possible. An unrecoverable error is not indicated until the error is retried the standard number of times.

The handling of track condition check is an exception to the general practice of restarting the channel program at the original start location. In this case, the home address and R0 record on the track is read to determine if the track is a defective or alternate track and to locate the corresponding alternate or defective track.

If it is an alternate track, the address of the defective track, plus one, is used in a seek command. (This is found in the ID field of the R0 record.) The operation is resumed after a search to the desired track position.

If it is a defective track, the address of the alternate track is used in a seek command. (This is found in the ID field of the R0 record.) The operation is resumed after a search to the desired track.

The direct access device error routine also handles both cylinder and head switching for sequential and split cylinder.

On a no-record-found error, no console message is issued if the arm has sought the correct track. Instead, a permanent error is flagged in the IOB.

If the device-dependent error routine is entered because of a channel data check, the error is handled exactly as if it was a device data check. If CCH is included in the system, and the routine is entered because of a channel control check or interface control check, the recovery procedure is the same as if the error was a seek check.

When the overflow incomplete bit is set, a CCW for the remainder of the record must be constructed, because only a partial record has been either read or written. The command code for the CCW to be constructed is taken from sense byte 5. Since it is required to construct the command outside of the original CCW list and then TIC to the original channel program, the data chaining must not extend beyond this CCW. Data chaining that includes a TIC may result in an uncorrectable overrun condition.

Whenever an overflow incomplete, in conjunction with a file mask violation, end-of-cylinder, or track condition check is detected, the direct access device error routine assumes that the continuation of the overflow record on the following track has the Record 1 in its ID field. This assumption is necessary because the error routine must orient the interrupted channel program to the first data record on the following track.

Sense Byte 0: The bits in the first sense byte have the meanings described below.

Bit 0: Command Reject

When this bit and bit 5 of sense byte 1 are both set, the write inhibit portion of the file mask has been violated; when it and bit 3 of byte 1 are both set, one of the following has occurred:

- A write command was not preceded by the necessary search or write command.
- A set file mask, reserve, or release command has been issued in a chain in which a previous set file mask was given.
- Head switching is being attempted in a command chain without a prior seek.
- Space count has been chained from a write command.
- A formatting command is being attempted after R0 on a defective track or following a record which is flagged as an overflow record.

When Command Reject occurs with Seek Check (byte 0, bit 7), one of the following has occurred:

- The 2841 has detected an invalid seek address. No seek is initiated.
- Less than six bytes of seek address were given. No seek is initiated.

When Command Reject occurs with none of the above, one of the following has occurred:

- An invalid command or a command associated with an uninstalled feature has been given.
- An invalid file mask has been given.
- The sum of key length plus data length exceeds $2^{16}-1$.
- An IPL command has been given after a set file mask has been issued.
- Command out has been presented in response to the request for the first byte of a write home address channel command word.

Bit 1: Intervention Required

This bit indicates that the specified file is:

- Not physically attached to the system.

- Physically attached to the system but not available for use because the file motor is not on, a cover interlock is open, etc.

Bit 2: Bus-Out Parity

This bit indicates that a data parity error has been detected during the transfer of information from the channel to the 2841. This check is an odd parity check and occurs on control, write, and search operations. This check is done by the 2841. A parity error detected during a command transfer is a bus-out check and not a command reject.

The sequence is repeated ten times if the error continues to occur. On the eleventh occurrence of the error condition, a message is written to the operator and the control unit is considered inoperative.

Bit 3: Equipment Check

This bit indicates that an unusual condition has been detected in the control unit or device. A message is written to the operator.

The conditions that cause the bit to be set are defined in sense byte 2, as follows:

1. Unsafe (Byte 2, Bit 0) - This bit is used to indicate that a file has given an unsafe indication. Some of the causes are:

   - More than one head has been selected.
   - The file is trying to read and write at the same time.
   - The write gate is off, and the write driver is on.
   - The write gate is on, and the write driver is off.
   - The erase driver is off, and the erase gate is on.
   - The erase driver is on, and the erase gate is off.
   - One of the DC file voltages has been lost (2311 only).

2. Serializer/Deserializer Check (Byte 2, Bit 2) - This bit indicates that a bit has been either lost or gained when the parallel channel data is converted to serial file data during a write operation.

3. 2841 Arithmetic Logic Unit (ALU) Check (Byte 2, Bit 4) - This bit indicates that the microprogram has detected impossible conditions, indicating an equipment malfunction. Some examples are:

   - Addition of nonzero quantities has resulted in a zero result with no carry.

   - Counting has been found to be not synchronous with cyclic check processing during a write operation.

   - File status decoding yielded self-contradictory results.

4. Unselected Status (Byte 2, Bit 5) This bit indicates that at least one of the file status lines is on without any file module being selected. This indicates a file malfunction of some kind since no line should be on prior to selection.

A message is written to the operator.

Bit 4: Data Check

This bit indicates that a data error has been detected in the information received from the file.

Data Check is never set in the key field when executing a read data command nor is it set in the Home Address (HA) when executing a read R0 command.

Data Check will usually occur when trying to read a record which overflows a track.

Bit 5: Overrun

This bit indicates that either a service-out signal was not received in the 2841 within the specified time allowed after service-in or that a chained CCW was issued but that it was received too late to be properly executed. Detection of an overrun during reading or writing causes an immediate stop of data transmission. When writing, the remaining portion of the record area is padded out with valid zeros. On reading or searching, indication is given as soon as it is detected.

The original sequence of operations is repeated ten times if the error continues to occur. On the eleventh occurrence of the error, a message is written and the task is abnormally terminated.

Bit 6: Track Condition Check

This bit does not apply to the 2301. It is set for other direct access devices whenever the following conditions are detected.

- Any single track command other than search HA, read HA, or read R0 is executed on a defective track. Note that after seeking to a defective track, it is necessary to execute a single track command if a track condition interrupt is desired; otherwise, none of the flag bytes on the defective track will ever be read and the interrupt cannot occur.

- Any multitrack command or overflow operation other than search HA, read HA, or read R0 switches to a defective track.

- Any multitrack command (including read HA, read R0, or search HA) or overflow operation attempts to switch from an alternate or defective track on which some read or search has been executed. Unless some flag byte has been "read" (in the current chain) by the 2841, the 2841 cannot "know" when it is switching from such a track. When such an interrupt occurs, head switching does not occur.

Bit 7: Seek Check

This bit indicates that the file has been unable to successfully complete a seek because:

- The transferred seek address is outside the valid address boundaries of the file. Command Reject is also set.

- Less than six bytes of seek address are sent. Command Reject is also set.

- A hardware failure which results in the access mechanism failing to detect correctly has occurred.

- The last three bytes of the Home Address of the track (C, H, H) are not identical with the last three bytes of the seek address.

Sense Byte 1: The bits of the second sense byte have the following meanings.

Bit 0: Data Check in Count Field

This bit indicates that a data error has been detected in a count field read from the file. Data Check, bit 4 in sense byte 0, is also set. Error detection is the same as described for Data Check. The operation is terminated at the end of the count field.

Bit 1: Track Overrun

This bit indicates that writing has not been completed by the time index point is detected. This type of error is created during a write R0, write count-key-data, write key-and-data, write data, or space count operation.

If a bad record thus written is read during subsequent read commands, Track Overrun will not be set. Data Check will be set if a data check occurs.

Bit 2: End-of-Cylinder

This bit indicates that the CCW command chain has not been completed, but that end-of-cylinder has been detected.

Bit 3: Invalid Sequence

This bit indicates that an attempt has been made to execute an invalid sequence of CCWs. Invalid sequences are normally related to write operations. The Invalid Sequence bit will also be set if two set file mask CCWs are attempted in the same chain of CCWs, if head switching is attempted in a command chain without a prior seek, or if space count was chained from a write command. Command Reject, Bit 0 of byte 0, is also turned on when an invalid sequence is encountered.

Valid sequences are defined in the individual command descriptions. A summary of valid sequences follows:

1. Write HA

2. Write HA
   Write R0
   Write count-key-data

3. Search HA
   TIC *-8
   Write R0
   Write count-key-data

4. Search ID Equal (R0)
   TIC *-8
   Write count-key-data
   Write count-key-data
   Write Special count-key-data

5. Search ID Equal (Previous ID)
   TIC *-8
   Write count-key-data

6. Search Key Equal (Previous Key)
   TIC *-8
   Write count-key-data (Rn)
   Search ID (Rn)
   Read key-and-data (Rn)
   Write count-key-data (Rn+1)

7. Search ID Equal
   TIC*-8
   Write key and data or Write Data

8. Search Key Equal
   TIC *-8
   Write Data

Bit 4: No Record Found

The No Record Found condition occurs whenever the following occurs:

- The 2841 senses two index points while performing a single track read or search operation other than read R0 or read HA.

- Either the home address or R0 cannot be found on a track. Byte 1, Bit 6, Missing Address Marker, is also set. This combination usually indicates that the 2321 failed to pick up a strip.

- A space count command is executed under different conditions.

Bit 5: File Protected

This bit indicates that a seek, write CCW, or M/T read or search command was issued that violates the file mask. The write file mask, if violated, also causes the command reject bit to be set.

Track Condition, bit 6 of byte 0, is set instead of the File Protected bit for M/T commands which attempt to switch from defective or alternate tracks.

Bit 6: Service Overrun (2301) or Missing Address Markers (2841)

The service overrun bit is turned on when a service out signal is not received in the 2301/2820 within the specified time allowed after the service in signal. The overrun bit (bit 5 of sense byte 0) is also set.

The incident of a missing address marker is detected during the execution of any command or chain of commands which operate successive count fields on a track.

The detection is accomplished by identifying that two successive records on a track have equal bit conditions in bit 0 of the flag bytes. In normal conditions, bit 0 of the flag byte is always a zero for all even numbered records and is always a one for all odd numbered records.

Upon detection of a missing address marker, bit 6 of sense byte one (missing address marker) is turned on for all commands or chained commands except search ID CCWs. The search ID CCW may be used to pass over the missing address marker so that the remaining data on the track can be retrieved.

When the missing address marker bit is set with bit 4 of byte 1, the no record found bit, it probably indicates failure to pick up a strip (2321).

The missing address marker bit is set with bit 7 of byte 1, the seek check bit, to indicate a 2321 ballast cell.

Bit 7: Overflow Incomplete

This bit is used with the optional record overflow feature. It is set as follows:

- Overflow to a bad track: overflow incomplete and track condition check bits are set.

- Overflow from an alternate track: overflow incomplete and track condition check bits are set.

- Overflow to a file protected boundary: overflow incomplete and file protected bits are set. (Track condition check is set instead of the File Protected bit for M/T commands which attempt to switch off defective or alternate tracks.)

- Overflow to wrong track: head number compares unequal: overflow incomplete and seek check bits are set.

- Overflow to end of cylinder: overflow incomplete and end-of-cylinder bits are set.

2250 Error Routine

The 2250 error routine attempts recovery from interface control checks and channel control checks. (See the flowchart, "General Operating Procedure, Graphics.")

The bits in the first sense byte have the meanings described below:

Bit 0: Command Reject

The setting of this bit indicates that there was an invalid modifier bit in a command, or an invalid command sequence. This is considered a permanent error.

Bit 2: Bus-Out Check

This bit is turned on when a parity error is detected on a command or data byte during bus-out. The operation is retried once, and on the second occurrence of the error condition a message is written to the operator.

Bit 4: Data Check

The setting of this bit indicates that a buffer parity error was detected either during a read operation or during image regeneration. The operation is retried once, and on the second occurrence of the error condition a message is written to the operator.

Bit 6: Buffer Running

This bit is set when orders are being executed in the buffer.

Bits 1, 3, 5, 7: Should Not Occur

If any of these bits are set, a catastrophic error is assumed. The Log Out bit in the IOB is set, and the permanent error procedure is performed.

2260 (1053) Error Routine

The 2250 error routine attempts recovery from interface control checks and channel control checks. (See the flowchart, "General Operating Procedure, Graphics.")

The bits in the first sense byte (byte 0) have the meanings described below:

Bit 0: Command Reject

The setting of this bit indicates that there was an invalid modifier bit in a command, or an invalid command sequence. This is considered a permanent error.

Bit 1: Intervention Required

The Intervention Required bit is set if a Write 1053 Printer command is encountered, and the 1053 Printer is not ready.

Bit 2: Bus-Out Check

This bit is turned on when a parity error is detected on a command or data byte during bus-out. There are three cases:

1. Bus-Out Check Initial Selection, command or data byte parity for 2260 or 1053.
2. Bus-Out Check Data Transmission, data byte parity for 2260 following Write Line Address or Erase command.
3. Bus-Out Check Data Transmission, data byte parity for 2260 following Write Buffer Storage command.

For 1 and 2, the operation is retried once, and on the second occurrence of the error condition a message is written to the operator.

For 3, an Erase Buffer Storage command is issued, and the Write command is reissued; on the second occurrence of the error condition a message is written to the operator.

Bit 3: Equipment Check

The setting of this bit indicates that a parity error was detected during a manual read input operation.

When this condition occurs, the 2260 error routine transfers control to its second load (load module IGE0110B), which erases the message in error. Anerror message is then displayed on the screen. No automatic retry is provided. If a retry is desired, the necessary coding must be included in the user's program. If the error is encountered a second time, the error should be considered permanent.

Bits 4-7: Should Not Occur

If any of the "Should Not Occur" bits are set, a catastrophic error condition is assumed. The Logout bit in the IOB is set, and the permanent error procedure is performed.

2280/82 Error Routine

The 2250 error routine attempts recovery from interface control checks and channel control checks. (See the flowchart, "General Operating Procedure -- Graphics.")

The IBM 2280/82 Standard Error Recovery procedures will attempt to retry and to recover from any errors which occur before execution of orders in the 2840 buffer. Any errors that occur subsequent to execution of device orders will not be retried by the error recovery module. This is primarily due to the fact that photographic film is not reuseable in the same way that magnetic disk, tape, or drum, are considered to be reuseable. Secondarily, the control unit and buffer programming capabilities are so extensive that the buffer program is capable of modifying itself while it is being executed. Since the Graphic Access Method has no facility to insure the reuseable condition of the buffer program once it has been started, recovery of error conditions encountered while using film will require each user to develop recovery procedures that best suit his application.

Sense Byte 0: The bits in the first sense byte have the meanings described below.

Bit 0: Command Reject

The setting of this bit indicates that there was an invalid modifier bit in a command, or an invalid command sequence. This is considered a permanent error.

Bit 1: Intervention Required

The setting of this bit indicates one of the following conditions:

a. Connect Recorder order was performed but the Recorder was not ready.
b. Connect Scanner order was performed but the Scanner was not ready.
c. A connected unit became not ready while Device End was outstanding.

Device End and Unit Check status will be presented to the control unit and the buffer program will be stopped. A standard intervention required message will be written to the console operator but when the device is made ready the IBM 2280/82 Error Recovery Module will not attempt to retry the I/O operation. The Error Recovery Module will return control to user by allowing the IOS to post a permanent error condition.

Bit 2: Bus Out Check

The setting of this bit indicates that a parity error occurred on a command or data byte. Unit Check status occurs on a command parity error. Device End and Unit Check status occurs on a data parity error. If Bus Out Check occurs, the IBM 2280/82 Standard Error Recovery Module will attempt to retry the I/O operation once.

Bit 3: Equipment Check

The setting of this bit indicates that equipment errors have been detected in the film unit. Equipment check errors cause the buffer orders to be stopped and set the Device End and Unit Check status bits. Since this error occurs subsequent to the execution of buffer orders, it will not be retried by the IBM 2280/82 Standard Error Recovery Module.

Bit 4: Data Check

The setting of this bit indicates one of the following conditions:

a. A buffer parity error occurred during a Read Buffer Command. This condition will cause Unit Check status to accompany Channel End and Device End. The IBM 2280/82 Error Recovery Module will attempt to retry the command once.
b. A buffer parity error occurred during the execution of the buffer order program. This condition will cause Unit Check to accompany Device End status and will not be retried by the IBM 2280/82 Error Recovery Module.

Bits 5, 6: Should Not Occur

If either or both of these bits are set, a catastrophic error is assumed. The Log Out bit in the IOB is set, and the permanent error procedure is performed.

Bit 7: Illegal Sequence

The setting of this bit indicates one of the following conditions:

a. The location specified by a Start command or a Transfer order did not contain a set mode byte (x'2A').

b. A fixed length control order (i.e., GCON) was not immediately followed by a set mode byte.

c. An odd address or count was specified by a Transfer class order, or a Load/Store class order.

Any of these conditions will stop the execution of buffer orders and will cause Unit Check to accompany Device End status. The IBM 2280/82 Error Recovery Module will not attempt to retry the I/O operation.

Sense Byte 1: The bits of the second sense byte have the following meanings.

Bit 0: Read Count Check

The setting of this bit indicates one of the following conditions:

a. The Read Byte count was less than 8 when a Read All End-Points order was executed.

b. The Read Byte count was less than 8 at the end of a 4-byte read operation in graphic mode.

c. The Read Byte count became zero while reading in stroke mode or when leaving character mode.

Any of these conditions will stop the execution of buffer orders and will cause Unit Check to accompany Device End status. The IBM 2280/82 Error Recovery Module will not attempt to retry the I/O operation.

Bit 1: Recorder Film Low

The setting of this bit indicates that the Recorder film supply is low. This bit will be set by each Film Advance order to the Recorder when the supply cassette indicator shows there is 10±5 feet remaining on the reel. Buffer order execution will be stopped immediately and Unit Check and Device End status will occur. The IBM 2280/82 Error Recovery Module will not attempt to retry I/O operation that cause this condition.

Bit 2: Recorder Forced Gap

The setting of this bit indicates that a film file gap was forced during the recording of a frame. Buffer order execution stops immediately and Unit Check and Device End occurs. The IBM 2280/82 Error Recovery Module will not attempt to retry the I/O operation.

Bit 3: Film Motion Limit (2282 Only)

The setting of this bit indicates that backward moving order was attempted on the Scanner when Column 1 is full or Column 2 is empty. Buffer order execution stops immediately and Unit Check and Device End occurs. The IBM 2280/82 Error Recovery Module will not attempt to retry the I/O operation.

Bits 3 (2280 only) and 4: Should Not Occur

If bit 3 is set during a 2280 error or if bit 4 is set during either a 2280 or 2282 error, a catastrophic error is assumed. The Log Out bit in the IOB is set and the permanent error procedure is performed.

Bit 5: 2840 Output Check

The setting of this bit indicates that a parity error in orders or data bound for the film unit was detected at the 2840 output bus. Buffer order execution stops immediately and a Device End and Unit Check occurs. The parity error is propogated to the film unit and causes an Equipment Check (sense byte 0, bit 3), to occur.

Bit 6: 2840 Input Check

The setting of this bit indicates that a parity error was detected at the input to the 2840 in data received from the film unit in response to a Read Device Register command. The command is not terminated. Unit Check, Device End and Channel End are set upon completion of the command. If this error occurs the IBM 2280/82 Error Recovery Module will attempt to retry the I/O operation once.

Bit 7: Graphic Check

The setting of this bit indicates that an overflow of the deflection system was caused by one of the following conditions:

a. A relative vector when overflow is disabled.

b. A space in Stroke mode or Variable-Space Character mode when overflow is disabled.

c. A new-line condition in Fixed-Space Character mode when the Y coordinate is less than the required new-line space and overflow is disabled.

Any of these conditions will stop the execution of buffer orders and will cause Unit Check to accompany Device End status. The IBM 2280/82 Error Recovery Module will not attempt to retry the I/O operation.

Common Routines

Common routines are used by the device-dependent error routines to analyze the type of error, to write console messages, and to update the statistics tables. These routines are:

- Error interpreter, which tests the sense bits and the CSW status bits to determine the type of error. This routine is resident; its load module name is IECINTRP.
- Write-to-operator routine, which constructs an "error" or "intervention required" message and writes that message to the operator. This routine is transient; its load module name is IGE0025C.
- Statistics update routine, which keeps the counters in the statistics tables up-to-date with the temporary and permanent errors determined by the device-dependent error routines. This routine is transient; its load module name is IGE0025D.

ERROR INTERPRETER ROUTINE

The error interpreter routine determines the types of errors for which error correction is required. The address of the routine is contained in the communication vector table (CVT).

The error interpreter "ANDs" the codes in the code and address constant list of the device-dependent routine to the corresponding sense or status bit. (See Section III: General Operating Procedure.) If the result is not zero, the bit is on and correction procedures for the specified sense or status condition must be taken. The address constant is added to the linkage register and an address within the device-dependent routine is obtained. Control returns to the device-dependent routine at that address.

WRITE-TO-OPERATOR ROUTINE

Device-dependent routines use the IOS write-to-operator routine (IGE0025C) to format and write any of three types of messages. These are the intervention-required message, the error message, and the inoperative-path message. The write-to-operator routine writes the appropriate message by issuing an SVC 35 instruction.

The intervention-required message (coded IEA000A) alerts the operator to needed action. Such action may consist of: making the I/O unit ready, clearing a card jam, inserting paper in the printer or printer-keyboard, etc.

The error message (coded IEA000I) is used when a permanent error has occurred. If the logout bit[1] in the input/output block is set, the write-to-operator routine passes control to the outboard recorder (IGE0025F). The outboard recorder constructs a record describing the error and writes the record on the SYS1.LOGREC data set.

The inoperative-path message (coded IEA001I) is issued when at least one, but not all, of the possible paths to a device is inoperative. The message is for the operator's information only, since an alternative path will be used if possible. If no path is operative, the intervention-required message is issued instead.

The message formats contain a standard message code, message text, channel and unit address, command code, status bits from the channel status word, and the sense bytes from the device's control unit. The job name and, when applicable, the volume serial number are included. For direct access devices, the cylinder and track addresses are also written. For tape, the block count is included. (For the exact message contents and meanings, refer to the Messages and Codes publication.)

STATISTICS UPDATE ROUTINE

The statistics update routine updates the 4-bit counters contained in the statistics table for the device-dependent routines. (See Introduction: Statistics Table.)

The work area of the statistics table entry for a particular device indicates the counters that must be updated. (See Section V: Statistics Table.) The first byte of the work area corresponds to the bits in sense byte 0. The second work area byte corresponds to sense byte 1, or, if sense byte 2 is significant for the device type, to bits from both sense bytes 1 and 2. The order of correspondence is device-dependent and indicated by the counts maintained in the statistics table entries shown in Section V.

The first (low-order) bit of the work area corresponds to the first 4-bit counter in a statistics table entry. This correspondence continues through the counters and the last bit of the work area.

When determining the counters to be updated, the statistics update routine inspects the work area starting at the last work area bit (high-order). The table

[1] The logout bit is bit 7 of the IOBFLG3 field, offset 8 of the IOB.

entry counters are thus updated from the right side of the table to the left side starting from the bottom. However, if a data check is indicated, the read and write counters are updated first.

If an overflow occurs in the counters, the statistical data recorder is entered for external error recording. (See following section.)

I/O Inboard and Outboard Recording Routines

The I/O inboard and outboard recording routines, the outboard recorder (OBR) and the statistical data recorder and channel-check recorder (SDR/CCR), are part of the recovery management facilities. They are provided as an extension of the IBM-supplied error routines. The OBR and SDR/CCR use the I/O supervisor transient area.

The OBR and SDR/CCR collect environmental and statistical data regarding errors that occur in I/O devices and control units. These errors are termed I/O outboard errors. The OBR and SDR/CCR format the data and enter it as an external error record on the system residence device.

The channel-check handler (CCH) collects and formats environmental data regarding a channel failure. This type of error is termed an I/O inboard error. SDR/CCR enters the record created by CCH on the system residence device.

THE OUTBOARD RECORDER

The OBR may be called by an IBM-supplied error routine whenever an I/O device has a permanent error. OBR is also called for a temporary or permanent channel failure. In the latter case, OBR tests the logout bit in the IOB to determine if an inboard record is to be entered on the system residence device. If so, OBR gives control to SDR/CCR. (See Table 2 for format of the logout bit.)

If an outboard recording, rather than a channel recording, is to be made, OBR constructs a record entry that contains a description of the device environment at the time of the failure and writes the record entry, and an end-of-file on the system residence device. OBR then passes control to SDR/CCR.

OBR Interface: When I/O outboard recording is required, the address of the request element associated with the error is passed to OBR in register 1. From the request element, OBR obtains the addresses of the UCB and the IOB. The UCB indicates the device in error and is also used to determine the address of the statistics table entry for the failing device. The IOB is used to determine such data as program identity, CCWs, and the CSW for the failing device.

The error routine which invokes the OBR loads the last 4 digits of the module name for OBR in a register. Control is routed to the task supervisor. The supervisor uses the digits in the register to complete the name of OBR, which is then loaded from the system residence device into the I/O supervisor transient area and given control. No other error routines can use the I/O supervisor transient area until OBR has completed its processing.

OBR terminates by passing control to SDR/CCR when its processing is completed. It can also be terminated by SER1 or the machine-check handler (MCH) before finishing its processing. When SER1 or MCH detects a machine check, it frees the request element associated with the error that OBR was processing. No I/O outboard recording is written in this case.

THE STATISTICAL DATA RECORDER AND CHANNEL-CHECK RECORDER

The SDR/CCR has two functions: (1) placing I/O inboard error records created by the channel-check handler (CCH) on the system residence device and (2) updating the statistics counters on the system residence device whenever one of the error statistics counters in the statistics table overflows.

When a temporary or permanent channel failure occurs, the SDR/CCR is called indirectly by the channel-check handler (CCH) via the statistics update routine if the CCH option was selected at system generation.

Whenever the statistics update routine (an extension of an IBM-supplied error routine) finds that one of the 4 bit error statistics counters within the statistics table for a device contains a count of 15, the statistics update routine calls the SDR/CCR. The SDR/CCR adds that 4-bit counter plus the partial counts within the other counters to the expanded statistics counters on the system residence device. Thus, a summary list of the various device failures is available for inspection.

SDR/CCR Interface: SDR/CCR is called in the manner described above in "OBR Interface." It is called either by an error routine when a statistics counter overflows or by the channel-check handler or OBR when a permanent error occurs. If SDR/CCR is called by an error routine to perform a statistical update, it resets the statistics table counters for future use. After its processing is completed, SDR/CCR returns to the supervisor via SVC 3.

Whenever the statistics update routine detects that a statistics counter has overflowed at the same time that a permanent error has occurred, it calls the OBR to perform the outboard recording. The OBR then calls the SDR/CCR, which performs the statistical update.

MODULE OPERATION

Three modules, all of which use the I/O supervisor transient area, make up the I/O inboard and outboard recording routines:

- The OBR module, load module name IGE0025F.
- The SDR/CCR module, load module name IGE0525F.
- The error message module, used by both the OBR and SDR/CCR, load module name IGE0425F.

The OBR module is executed whenever a permanent error occurs. The SDR/CCR module is executed whenever a temporary or permanent channel error occurs or whenever a counter reaches the value of 15. If an error occurs during module execution, the error message module is called.

After an error routine determines that I/O inboard or outboard recording is required, it loads the name of either the OBR or SDR/CCR for the transfer control (XCTL) routine of the contents supervisor, and branches to the XCTL routine.

After being loaded, the OBR or SDR/CCR module gains control from the common contents supervision subroutine, which masks all interruptions as disabled. The module is executed and then brings the next required module into the I/O supervisor transient area via the XCTL routine. The last module to be executed issues an SVC 15 and an SVC 3 (EXIT) on completion, and control returns to the supervisor.

Depending upon the type of system residence device and the number of UCBs in the system, the SER routine requires a data set extent, on the system residence device, of at least two tracks. These tracks are formatted and initialized by the SYS1.LOGREC initialization program.[1]

The extent is termed the SER output data set and contains in contiguous positions:

- Header record
- Statistical data records
- Record entries

A detailed description of these records may be found in Appendix D.

Note: For both statistical data records and outboard records, a 2314 direct access storage facility is considered to be nine devices. The devices are differentiated by the physical location returned in the fifth sense byte. This value does not necessarily correspond to the device address in the logical unit address.

If the control program configuration is MFT or MVT with SER1 or MCH, the OBR and SDR/CCR modules use a 256-byte area appended to the SER resident DCB as a work area. For any other control program configuration, (PCP or MFT or MVT without SER or MCH), the OBR and SDR/CCR use the machine logout area (decimal locations 128 through 383).

If a machine check occurs, SER1 or MCH gains control. They simulate the termination of OBR and SDR/CCR. Priority is given to any machine check occurring during the operation and only that CPU check is recorded. To read and write the header record, statistical data records, and record entries, the OBR and the SDR modules use the EXCP macro instruction. The IOB is loaded with the DCB, ECB, and CCW list addresses. For error messages, the error message module uses the WTO macro instruction. Before these macro instructions are issued, however, the system is enabled.

The channel programs necessary for operations upon the SER output data set consist of the following CCWs:

- Read home address (required only by the SDR).

- Search on ID or key equal for the appropriate record.

- TIC to the first CCW until the record is found.

- Read or write the record.

After execution of the channel program and when the I/O request is complete, the modules inspect the ECB completion code, which provides the status of the execution as follows:

- Normal. The record has been read into the log out area or written onto the system residence device with no errors.

----

[1] For information concerning this program, refer to the IBM System/360 Operating System: Utilities, Program Logic Manual, Form Y28-6614.

- Intercepted. The IOB for the request was intercepted for other error recovery procedures because of an error at a previous device end; the EXCP macro instruction is reissued.

- Fail. The request has failed because either the data set is not within the extent specified or a permanent I/O error has occurred; an error index value and the load name for the error message module is set and an SVC 3 instruction is issued.

  Note: An SVC 3 instruction is also issued when the header safety byte is zero.

The function of each module follows. The flow among them is shown in Chart 17.

OBR Module

Upon gaining control, the OBR module tests a flag in the SER resident DCB. If the flag is on, OBR uses the 256-byte area appended to the SER DCB for a work area. If the flag is off, OBR uses the machine logout area. OBR then reads the header record from the system residence device into the area selected. OBR formats and writes the following fields on the system residence device:

- Record entry label
- Record entry type
- Device type
- Date
- Program ID
- First CCW
- Failing CCW
- CSW
- Sense information
- Channel and device address
- Last seek address
- Volume label
- Time

OBR then writes an end-of-file, updates the header record, and writes the updated header record on SYS1.LOGREC. The name of the SDR/CCR module is loaded into the XCTL register and OBR branches to the XCTL routine.

Whenever an error occurs (for example, when the records to be written do not fit on the extent) OBR calls the error message module by loading its name into the XCTL register and branches to the XCTL routine.

SDR/CCR Module

The SDR/CCR module first determines if a CCR recording is to be written. (It checks the SYS1.LOGREC DCB for the presence of the channel-check handler in the system.) If a CCR recording is to be made, SDR/CCR writes all channel recordings, writes an end-of-file, and updates and writes the header record on the system residence device. If necessary, SDR/CCR calls the error message module. A check is made to find out if SDR recording is to be made. If not, SVC 15 is issued to free the request element and control returns to the task supervisor via an SVC 3.

If an SDR recording is to be made, the program reads the statistical data records for the particular device from the system residence device and obtains the internal statistical counters which are to be added to the external counters in the statistical data record. SDR/CCR then adds the appropriate statistical table entry counters to the external statistical data record counters and writes the record back onto the system residence device. If necessary, SDR/CCR calls the error message module; otherwise, it issues SVC 15 to return the request element to the freelist, and an SVC 3 instruction to return control to the task supervisor and, from there, to the I/O supervisor.

Error Message Module

This module handles all errors occurring within the I/O inboard and outboard recording routines and writes a message to the operator about them. If the OBR routine was in control when the error occurred, the error message module calls in the SDR/CCR module. If the SDR/CCR module was in control when the error occurred, the error message module returns to the task supervisor.

The following error messages may be written to the operator:

1. IFB001I I/O ERROR DURING SER
2. IFB003I SER RECORDING AREA FULL
3. IFB004I SER DISK AREA FORMAT ERROR

SYS1.LOGREC DATA SET STATISTICS UPDATE ROUTINE

The SYS1.LOGREC data set statistics update routine, IFBSTAT-IGC0007F, is a transient SVC routine (SVC 76). It adds counts of recoverable errors maintained by the I/O supervisor error routines in the statistics table ("Section V: Control Block and Table Formats") to the statistical data on the SYS1.LOGREC data set. The routine is entered when the operator issues a HALTEOD (halt-end-of-day) command.

Section IV: SVC Transient Area Routines

The SVC transient area routines are type 3 SVC routines that are called into the SVC transient area in response to system macro instructions. The macro instruction names, load module names, and SVC numbers are:

- PURGE: Load module IGC0001F and IGC0101F (SVC 16)
- RESTORE: Load module IGC0001G (SVC 17)
- DEVTYPE: Load module IGC0002D (SVC 24)
- IOHALT: Load module IGC0003C (SVC 33)

SVC PURGE ROUTINE

The purge routine removes (purges) request elements that are associated with a task or a data set. Depending on the options specified in the purge input parameter list, request elements can be purged either from the I/O supervisor logical channel queues only, or from all of the following queues: the asynchronous exit queue of the task supervisor, the request blocks chained to the TCB, and the logical channel queues.

If a request for a shared direct access device is to be purged, and the stand-alone seek has been issued, but the data transfer operation has not begun, the request element is not dequeued. Instead, the purge routine turns on the HIO flag in the UCB (bit 0 of the UCBHA field). This allows the I/O supervisor to release the device.

Request elements are purged according to the options specified in the purge option byte of the input parameter list. The format of the parameter list is:

Word 1
    Byte 1
        Purge options (see below)

    Bytes 2, 3, and 4
        DEB address

Word 2
    Byte 1
        Completion code

    Bytes 2, 3, and 4
        TCB address

Word 3
    Byte 1
        Count field for quiesce option contains the number of request elements whose I/O operations have not completed.

    Bytes 2, 3, and 4
        Address of the initial link field for chaining IOBs to be purged. The initial link field can be the user purge field in DEB, DEBUSRPG, or any area the user selects. The initial link field points to the first IOB in the chain. (The IOBRESTR field of the last IOB contains X'FF' in the low order byte.) Figure 7 illustrates the IOB chain.

        No chaining is done when the TCB/HALT I/O option is specified.

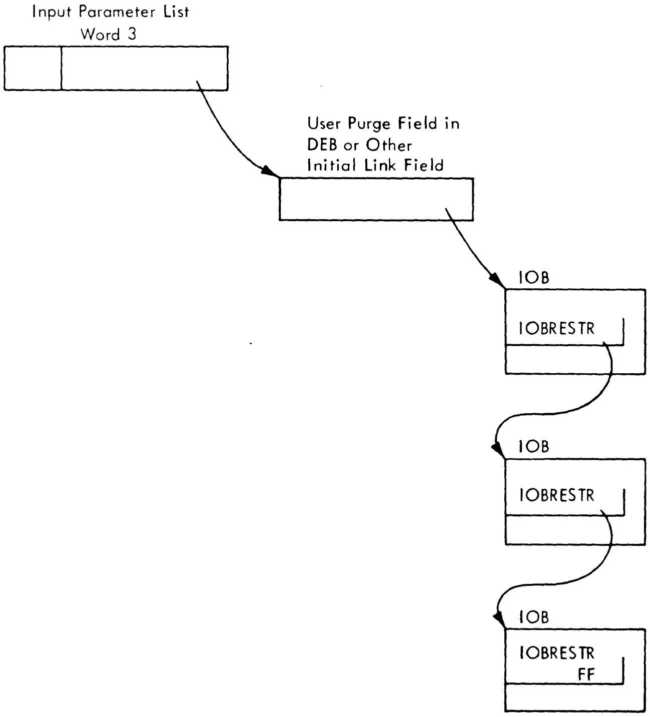


Figure 7. IOB Chain For Purge

OPTIONS BYTE (Word 1, Byte 1 above):

Bit 0 - Specified DEB or DEB chain.

= 0 - Purge request elements associated with complete DEB chain starting at the DEB specified in bytes 2, 3, and 4 of word 1.

= 1 - Purge only the request elements associated with the DEB specified in bytes 2, 3, and 4 of word 1.

Bit 1 - POST requests purged or ignore posting.

= 0 - Do not POST the purged requests.

= 1 - POST the purged requests. ECB Completion code = X'48'.

Bit 2 - HALT I/O or quiesce active requests.

= 0 - Allow the active requests to quiesce.

= 1 - HALT the I/O operations. (The HALT I/O is simulated if the operation is a seek.)

Bit 3 - Purge all requests or purge related requests only.

= 0 - Purge all requests.

= 1 - Purge only related requests.

Bit 4 - (Not used)

Bit 5 - Purge all queues or bypass RB purge.

= 0 - Purge AEQ, RB, and I/O supervisor logical channel queues.

= 1 - Purge only the I/O supervisor logical channel queue(s).

Bit 6 - Purge by TCB or purge by DEB.

= 0 - Purge by DEB.

= 1 - Purge by TCB.

Note: This bit must be zero in order to honor bit 0. If this bit is one, all requests associated with the TCB are purged, and bit 0 is ignored.

Bit 7 - (Not used)

After the request element is purged, the SVC purge routine returns the element to the freelist, and if the element is not active, dequeues it from the logical channel queue. If the quiesce and/or DEB option is specified, the non-active IOBs addressed by the purged request element are placed in the chain specified by the input parameter list (Word 3). (See restore routine.)

Note: If purging by DEB with the HIO option, all IOBs are chained. If purging by TCB with the HIO option, no IOBs are chained. If purging by TCB or DEB with the QUIESCE option, only the inactive IOBs are chained.

The following procedures are used to purge the logical channel queues of the I/O supervisor:

1. If purging by DEB (option bit 6 = 0), the SVC purge routine examines its extent(s) for the device(s) where the data set resides, and purges the associated logical channel of requests for the DEB. Option bit 0 determines if requests associated with a chain of DEBs or requests associated with a specific DEB only are to be purged.

2. If purging by TCB (option bit 6 = 1), the SVC purge routine examines the request element table and removes each active request element associated with the TCB from its corresponding queue in the following manner:

   a. Find the requests which are not on the freelist, starting at the beginning of the table.

   b. Get the logical channel address associated with the request.

   c. Purge all requests on the logical channel associated with the device. Repeat a through c until the request element table is exhausted.

If the quiesce option is specified, two counters are incremented for those active requests awaiting completion:

1. A main purge counter in the parameter list. (word 3, byte 1).

2. A sub-counter in each associated DEB.

Note: When the purge routine is entered, an ENQ macro instruction is issued to ensure exclusive use of the DEBs by the issuing task. A DEQ macro instruction is issued when all requests have been quiesced and/or purged. The resource names for the ENQ, DEQ macro instructions are:

Qname=SYSIEC16
Rname=F0

When all requests are processed, the main purge counter in the parameter list is tested for a count. If there is no count, a X'7F' is placed in the completion code byte of the parameter list and the SVC purge routine issues an SVC 3 instruction.

If there is a count, the routine issues a WAIT macro instruction to the purge ECB (word 2). When the WAIT is satisfied, i.e., when the purge ECB is posted by the purge complete subroutine, the SVC purge routine again scans the system for outstanding requests to purge according to the original options. The scanning continues until there are no more requests to quiesce. (See purge complete subroutine.)

The SVC purge routine contains six subroutines that purge the queues according to the options specified. These subroutines are:

- Asynchronous exit queue (AEQ) purge subroutine. Purges the asynchronous exit queue via the check subroutine.

- Request block (RB) subroutine. Purges the RB list via the check subroutine.

- Check subroutine. Purges the AEQ and RB list according to the options specified. Uses the validity check subroutine to determine if the TCB or DEB is valid for the request.

- Validity check subroutine. Determines if the purge option is for TCB or DEB and performs validity checking accordingly.

- DEB purge subroutine. Obtains the UCB address from each extent in the DEB. Uses the UCB purge subroutine to purge the logical channel associated with the extent. Determines if the purge option specified one DEB or a chain of DEBs.

- UCB purge subroutine. Purges the logical channel queue of the request elements associated with either the DEB or TCB specified by the option byte. Uses the validity check subroutine.

Purge Complete Subroutine

Whenever the SVC Purge is entered with the option to quiesce, the SVC purge routine must allow the active I/O requests to quiesce.

The SVC purge routine keeps a main counter in the purge parameter list (word 3, byte 1) for all active I/O requests. It also keeps a sub-counter in each DEB for all active requests associated with that DEB. Whenever SVC purge increments the counters, a pointer to the purge parameter list is stored in the DEB Purge ECB field, DEBECBAD.

The purge complete subroutine gets control when the I/O interruption supervisor, at completion of an I/O request, determines that an address is stored in the DEBECBAD field. The purge complete subroutine decrements the counters for each valid request that is completed. As each DEB counter is zeroed, the DEBECBAD field in the DEB is also zeroed. When the main counter in the parameter list is zero, the subroutine posts the ECB for the purge request complete, via the post routine interface. If the main counter is not zero, control returns to the I/O interruption supervisor.

RESTORE ROUTINE

The restore routine re-introduces purged requests (IOBs) to the system. Because the requests are strictly for I/O operations, the EXCP macro instruction is issued for each previously chained request. Register 1 contains a pointer to the address of the first IOB to be restored. Subsequent IOBs are chained through the IOBRESTR field.

The RESTORE macro instruction requires that the IFLGS field in the DCB be initialized in the same way as for the EXCP macro instruction.

DEVTYPE ROUTINE

When SVC 24 is executed, the DEVTYPE routine is loaded into the SVC transient area and receives control. The expansion of the DEVTYPE macro instruction places parameters in general registers; the DEVTYPE routine uses the parameters to identify the device, and to place device information in an output area.

The DEVTYPE routine looks at the parameter that specifies the address of a field that contains a DD name. Using that DD name, the routine finds the corresponding TIOT entry, and uses it to locate the proper UCB. The DEVTYPE routine then moves the UCB device code field to the high order word of the output area specified by the area parameter.

If the DEVTAB parameter is specified in the DEVTYPE macro instruction, and the device is a direct access device, the output area must be 5 words long. The DEVTYPE routine locates the device characteristics table via the communications vector table, moves the appropriate entry into the low order three words of the output area, and moves the blocksize field to the second word of the output area.

If the DEVTAB parameter is not specified, the output area must be 2 words long. The DEVTYPE routine determines the maximum block size from the device characteristics table (if the device is a direct access device) or from internal constants, and moves it to the second word of the output area.

When processing is complete, the DEVTYPE routine places a return code in general register 15. An error return code of 04 indicates one of the following conditions:

- No output area specified: the area parameter is missing from the DEVTYPE macro instruction.
- DD name not found: there is no TIOT entry that corresponds to the DD name supplied.
- Invalid UCB unit type field: the UCB unit type field (byte 4 of the device code field) does not specify a direct access, tape, or unit record device.

If the request is completed satisfactorily, the DEVTYPE routine places a return code of 00 in general register 15, and exits via SVC 3.

IOHALT ROUTINE

When the IOHALT routine (SVC 33) receives control, it is given the address of the UCB associated with the device to be stopped. It checks that address for validity, then inspects the UCB device code field to make sure that the device is not a direct access device.

IOHALT branches to a resident I/O Supervisor subroutine to issue the HIO instruction and examine the resulting condition code. If no error occurs, control returns to IOHALT.

The appropriate condition code is placed in general register 15. If the operation was successful, the IOHALT routine places a post code (X'48') in the ECB code field of the IOB and issues SVC 3 to exit.

have been loaded, issues return code 0 and returns.

If neither the LOADONLY or NOVERIFY option is specified, IEHUCSLD opens the printer DCB for BSAM. It skips the printer to the next page and prints a header line that specifies the unit, type ID, and mode (folding or non-folding). Then IEHUCSLD spaces two lines and prints two 120 character lines to display the images it has loaded into the 2821 generator storage.

If the header line requires images that were not supplied by the user, and the reset block data check mode is specified in the printer DD statement, the IEHUCSLD program does not space two lines after the header. If the user does not specify reset block data check mode in his printer DD statement, the space will occur; in either case the images that were not supplied will print as blanks.

When the three lines have been printed, IEHUCSLD skips the printer to the next page and tells the operator to check the images, using message IEH501A.

The operator must reply, 'OK' or 'NG'. If the reply is 'NG' the images are printed once more, and the operator is again requested to check the images. A second 'NG' reply causes the program to close the printer DCB and return with code 4.

If the reply is 'OK', IEHUCSLD closes the printer DCB, loads return code 0, and returns.

CHANNEL AVAILABILITY TABLE

The Channel Availability Table is created only if the M65MP option is selected at system generation. It is used by the I/O Supervisor to determine if a channel of the second CPU is available.

The Channel availability Table has the following characteristics:

- Creation. The Channel Availability Table is created at system generation, and initialized by the Nucleus Initialization Program.
- Storage Area. The table resides in the Prefixed Storage Area (PSA), which means each CPU has its own Channel Availability Table.
- Size. The table consists of two bytes for each of seven possible channels, no matter how many channels are actually configured in the system. For convenience, other fields located with the table in the PSA are shown in the format below.
- Means of Access. The entry for each channel is addressable by a label consisting of the characters "CHANn" where "n" is the number of the channel.

Hex	Dec				
0	0	MPCVT ADDRESS		CHAN0	CHAN1
8	8	CHAN2	CHAN3	CHAN4	CHAN5
10	16	CHAN6	reserved	OTHER PSA	
18	24	IOCPUID		CPUID	reserved
20	32	SHOULDER TAP MASK		reserved	

Figure 9. Channel Availability Table Format

MPCVT ADDRESS (4 bytes)
Pointer to the M65MP extension to the Communications Vector Table.

CHAN0 - CHAN6 (2 bytes each)

Byte 0

Bit 0: Channel busy. Set to 1 following SIO by the EXCP Supervisor, and set to 0 by the Interruption Supervisor after channel end.

Bit 1: Channel not operational. Set to 1 by the Nucleus Initialization Program (NIP) if the channel responds "not operational" (cc = 3) to Test Channel. This flag is also set to 1 following a VARY CHANNEL command placing the channel offline.

Bit 2: Channel not in system. Set to 1 by NIP if the channel is not included in system generation.

Bit 3: Channel not available. Assembled as a 1, and set to 0 by NIP following a Test Channel response indicating availability. This flag is also set to 1 following a VARY CPU command placing the channel's CPU offline.

Bit 4: Not used.

Bits 5-7: The physical channel address, from 0 through 6.

Byte 1
Not used.

OTHER PSA (4 bytes)
A pointer to the Prefixed Storage Area (PSA) of the other CPU.

IOCPUID (4 bytes)
Set by NIP to X'00000000' for CPUA and X'00000008' for CPUB. Used by EXCP Supervisor to set bit 4 of the M65MP flags in the UCB following SIO. The Interruption Supervisor uses this field to test the UCB flags following

an interruption. If the setting does not match, the interruption is identified as the second of two device ends presented when a device changes status from "not ready" to "ready." Some control units present one device end for this change to each attached channel.

CPUID (1 byte)
An identification byte set by NIP to X'C1' (= C'A') for CPUA and to X'C2' (= C'B') for CPUB.

STMASK (4 bytes)
Bit settings in the issuing CPU's Shoulder Tap Mask word are tested by the receiving CPU following a shoulder tap to see what actions are requested. In each case, the performing CPU resets the flag to 0 when the action is completed.

Bit 0: A Write Direct instruction has already been issued, but not accepted by the other CPU. Additional bits may be set "on" without issuing another Write Direct instruction.

Bit 6: Halt I/O requested.

Bit 16: A QUIESCE command is pending.

Bit 17: A VARY CPU command is pending.

Bit 24: I/O for Channel 0.

Bit 25: I/O for Channel 1.

Bit 26: I/O for Channel 2.

Bit 27: I/O for Channel 3.

Bit 28: I/O for Channel 4.

Bit 29: I/O for Channel 5.

Bit 30: I/O for Channel 6.

CHANNEL TABLE

The I/O supervisor uses the channel table to find the proper channel search module and to re-enable for additional I/O interruptions.

The channel table has the following characteristics:

- Creation. The channel table is created at system generation time.
- Storage Area. The table resides, as a permanent part of the resident supervisor, in protected resident storage (when protection is available).
- Size. The table contains one 4-byte entry for each physical channel attached to the system.
- Means of Access. The address of a particular physical channel is multiplied by 4 and added to the starting address of the channel table to obtain the address of the entry for that physical channel.
- Format. The format of the channel table is shown in Figure 10.

000 (Multiplexor channel)	Channel Search Module Address	Channel Mask	Unused
001 (Selector channel 1)	Channel Search Module Address	Channel Mask	Unused
002 (Selector channel 2)	Channel Search Module Address	Channel Mask	Unused
etc. . . .	. . .	. . .	. . .
	<------------2 bytes------------>	<---1 byte---->	<---1 byte--->
	<--------------------------4 bytes-------------------------->		

Figure 10. Channel Table Format

Channel Search Module Address (2 bytes)
This 2-byte field contains the address of the channel-dependent channel search module associated with the physical channel. A channel search module exists for each physical channel and it determines the next request to be started on that channel when it becomes available.

Channel Mask (1 byte)
This byte contains the channel re-enable mask.

DATA CONTROL BLOCK

A data control block (DCB) is used primarily by the open and close routines of data management, but the I/O supervisor is concerned with several of its fields.

The characteristics of a DCB are as follows:

- Quantity. There is one DCB for each data set.
- Creation. Space for the DCBs is reserved at problem program assembly time. The DCB is completed at open time.
- Storage Area. All DCBs reside in unprotected problem program storage.
- Size. Refer to IBM System/360 Operation System: System Control Blocks for the size and description of the entire DCB.
- Format. The DCB fields used by the I/O supervisor are shown in Figure 11.

FIELD	CONTENT	SIZE	SECTION
IFLGS	I/O Supervisor error flags	1 byte	Foundation Block
DEBAD	DEB address	3 bytes	
BLKCT	Tape block count	4 bytes	Tape

Figure 11. Data Control Block Fields Used by the Input/Output Supervisor

IFLGS (1 byte)
These are flag bits used by the I/O supervisor in communicating error conditions and in determining corrective procedures. They decode as follows:

Bits 0 and 1: DCB Exception

These two bits are used during error recovery procedures when accessing sequential files where positioning is important. The flags indicate the error status for the pertinent data set, and must always be present. They have the following value:

00 Normal conditions are present.
01 Error correction is in progress.
11 A permanent error condition exists.

Bits 2 and 3: Printer Overflow

These bits are set by error routines when printer overflow occurs. They have the following value:

10 Channel 9 printer overflow.
01 Channel 12 printer overflow.

Bits 4 and 5: I/O Supervisor Error Key

This key is mandatory and indicates whether or not an IBM-supplied error routine should be entered. The key is checked by the error routine interface. The flags have the following value:

00 Always use IBM-supplied error routines.
11 Never use IBM-supplied error routines.

Bits 6 and 7: Reserved

DEBAD (3 bytes)
This DEB address field contains the actual address of the data set's associated data extent block. The DEB contains the location and boundaries of that data set. The DEB address must always be present in the DCB foundation block after open time.

BLKCT (4 bytes)
This device-dependent area is used by the I/O supervisor to accumulate the tape block count for a data set on tape that is either being read or written. The block count designates physical unit position, not relative data set position. At the normal completion (channel end/device end) of all I/O requests, the IOB block count increment amount is added to this DCB block count field to form the updated tape block count.

Notes: An incorrect block count field results when an error occurs in the middle of a command chained channel command word list. In this case, it is the user's responsibility to adjust the DCB block count field. He must determine which CCW had the error and then decrement the block count so that it reflects the true count.

When an uncorrectable error occurs, the block count field is updated as usual. The effect of a halt I/O request by the purge routine on the block count field is unpredictable, since the I/O interruption condition is determined by the disposition of the device and control unit, which are variable.

DATA EXTENT BLOCK

A data extent block (DEB) describes the location and boundaries (extent) of a data set and points to the pertinent UCB(s) for the device(s) upon which the data set resides. The characteristics of a DEB are as follows:

- Quantity. There is one DEB for each DCB.

- Creation. The DEBs are created by the open routine.

- Storage Area. All DEBs reside in the dynamic area.

- Size. Refer to IBM System/360 Operating System: System Control Blocks for the size and description of the entire DEB.

- Format. The DEB format is shown in Figure 12.

	|<-------------	---4 bytes---	------------->|		
-16					
					PREFIX
- 4					
0	NMSUB	TCB ADDRESS			BASIC
+ 4	AM LENGTH	NEXT DEB ADDRESS			
+ 8	OPEN FLAGS	IRB ADDRESS			
+12	OPEN FILE TYPES	SYSTEM PURGE CHAIN			
+16	NMEXT	USER PURGE CHAIN			
+20	PRIORITY	PURGE LIST ADDRESS			
+24	PROT TAG | DEB ID	DCB ADDRESS			
+28	EXTENT SCALE	APPENDAGE VECTOR TABLE ADDRESS			
+32	MODE SET BYTE	UCB ADDRESS		Tape or Unit Record	DEVICE DEPENDENT
+32	FILE MASK	UCB ADDRESS		Direct Access	
+36	B B		C (START) C		
+40	H (START) H		C (END) C		
+44	H (END) H		NMTRK		
					ACCESS METHOD DEPENDENT and SUBROUTINE SECTION

The I/O supervisor does not use the shaded fields.

Figure 12. Data Extent Block Format

TCB ADDRESS (3 bytes)
This field contains the address of the TCB associated with this DEB.

NEXT DEB ADDRESS (3 bytes)
This field contains the address of the next DEB in a chain of DEBs running under the same task.

IRB ADDRESS (3 bytes)
The interruption request block (IRB) address for system error exits is contained here.

SYSTEM PURGE CHAIN (3 bytes)
This field contains the address of the first IOB in the chain of IOBs for I/O requests purged by the I/O purge routine.

NUMBER OF EXTENT ENTRIES (1 BYTE)
This byte contains the number of extent entries specified in this DEB.

USER PURGE CHAIN (3 bytes)
This field contains the address of the first IOB in the chain of IOBs for I/O requests purged by the purge routine for SVC 16.

PRIORITY (1 byte)
This field contains a binary number, assigned by the open routine, which indicates the dispatching priority of the task associated with an I/O request.

PURGE LIST ADDRESS (3 bytes)
This field contains the address of a parameter list for an SVC purge request. The I/O supervisor uses this field when purging to locate the appropriate purge ECB to post complete.

PROTECTION TAG (4 bits)
These bits contain the protection tag for the associated task.

DEB ID (4 bits)
These four bits identify this block as a DEB by means of a hexadecimal "F".

DCB ADDRESS (3 bytes)
The address of the DCB corresponding to this DEB is contained here.

EXTENT SCALE (1 byte)
This byte contains a shift factor used to determine the size of the extent entries in the DEB. This field contains 4 for a direct access device and 2 for a non-direct access device.

APPENDAGE VECTOR TABLE ADDRESS (3 bytes)
This field contains the address of the appendage vector table that is constructed at open. The appendage vector table consists of address constants (adcons) as follows:

0	DC A(EOE)	End of Extent
4	DC A(SIO)	Start I/O
8	DC A(PCI)	PCI
12	DC A(CE)	Channel End
16	DC A(XCE)	Abnormal End
	|<-4 bytes->|	

The I/O supervisor uses the respective adcon as a branch address to the five types of appendages. If an appendage is not used, the respective adcon points to a branch back to the I/O supervisor "normal" processing return.

When no appendages are required, this table contains five adcon's pointing back to the "normal" processing return.

FILE MASK or MODE SET (1 byte)
This field contains either the file mask for a direct access device or a mode set byte for tape.

UCB ADDRESS (3 bytes)
This field contains the address of a UCB associated with this DEB.

BB (2 bytes)
This field contains the bin number of a direct access volume (data cell drive).

CC START (2 bytes)
This field contains the cylinder address for the start of an extent on a direct access volume.

HH START (2 bytes)
This field contains the read/write head number for the start of an extent on a direct access volume.

CC END (2 bytes)
This field contains the cylinder address for the end of an extent on a direct access volume.

HH END (2 bytes)
This field contains the read/write head number for the end of an extent on a direct access volume.

NMTR (2 bytes)
This field contains the number of tracks allocated to a given extent on a direct access volume.

DEVICE TABLE

The device table is used by the I/O supervisor to locate the device-dependent start I/O, enqueue, or trapcode module that pertains to the device in use.

The following characteristics are pertinent:

- Creation. The device table is created at system generation time.
- Storage Area. The device table resides, as a permanent part of the resident supervisor, in protected resident storage (when protection is available).
- Size. Each entry in the device table is 6 bytes in length. There is always one entry per device type per enqueuing option specified for the device types at system generation time. For example, direct access devices using priority enqueuing would have one entry, direct access devices using FIFO queuing would have one entry, etc.
- Means of Access. The DEVTAB byte in the unit control block (UCB), when added to the starting address of the device table, gives the address of the proper entry in the table for the device.
- Format. The format of a device table entry is shown in Figure 13. The possible entries in the device table and their order, are:

```
Unit Record
     Priority
     FIFO
Tape
     Priority
     FIFO
Direct access (Disks)
     Priority
     FIFO
Direct access (Drums)
     Priority
     FIFO
Telecommunications
     Priority
     FIFO
Graphics
     Priority
     FIFO
```

Address of Enqueue Module	Address of Start I/O Module	Address of Trapcode Module
<--------2 bytes-------->	<---------2 bytes--------->	<---------2 bytes-------->
<------------------------	-----------6 bytes---------	------------------------>

Figure 13. Device Table Entry Format

ERROR RECOVERY PROCEDURE INTERFACE BYTES

Error recovery procedure interface bytes are produced by CCH to facilitate retry of a failing channel operation by the appropriate device dependent error routine. The format of the ERPIB is shown in Figure 14.

Byte 0	1	2	3	4	5	6	7
Multi-processing Ind. bits	UCB Address			Program Flags bits 0-3, 5-7	Not Used	Validity Ind. bits 3-5, 7	Term/Retry bits 0, 1, 5-7

Figure 14. Error Recovery Procedure Interface Bytes Format

Error Recovery Procedure Interface Bytes Field Definition

- Multiprocessing Indicators (Byte 0)

  Bits 0, 1, 2 - Unused.

  Bit 3 - Multi-System Feature present.

  Bit 4 - ID of failing CPU.
  If CPU1, bit 4 is zero.
  If CPU2, bit 4 is one.

  Bit 5 - Unused.

  Bits 6,7 - CPU status:
  00 - Shared
  01 - Partitioned
  10 - Shared, but reconfigured out.

  Note: If Multi-System Feature is not present, byte 0 of the ERPIB contains all zeros.

- UCB Address Pointer (bytes 1, 2, 3). This field contains the UCB address for the failing channel/unit used to locate the correct interface bytes for the current error.

- Program Flags (byte 4). This field contains program flags indicating the selection or interrupt sequence at the time of storing the CSW, etc.

  Bit 0 - SIO instruction stored a CSW.

  Bit 1 - I/O interrupt or TIO to an I/O interrupt stored a CSW.

  Bit 2 - TIO with no pending interrupt stored a CSW.

  Bit 3 - HIO stored a CSW.

  Bit 4 - Reserved.

  Bit 5 - Sense Data stored.

  Bit 6 - Valid count appears in CSW.

  Bit 7 - Unconditional No-Retry.

- Validity Indicators (byte 6). This field contains validity indicators. Note that a particular bit may indicate that its associated field is invalid even though the field may contain some useful information. The error routine normally should not use any field whose contents have been declared invalid.

  Bits 0, 1, 2 - Unused.

  Bit 3 - Retry code valid.

  Bit 4 - Unit status valid. The unit status has been indicated with correct parity.

  Bit 5 - Command address valid. The CSW contains a valid command address. This field minus eight bytes (CA-8) is the address of the CCW for the channel at the time of the channel error.

  Bit 6 - Unused.

  Bit 7 - Device address valid. A valid device address was indicated and appears in the I/O old PSW. The bit is set for reference by I/O supervisor and should not be used by the error routine.

- Termination code (byte 7, bits 0, 1). This field contains a termination code which specifies the termination sequences (signals) used on the I/O interface after the channel has detected an error. The bits are interpreted as follows:

  00 - Interface Disconnect (HIO sequence).
  01 - Forced Ending Sequence (STOP). The channel responded with Command Out to Service In after an error was detected.
  10 - Selective Reset.
  11 - System Reset.

- Retry Code (byte 7, bits 5, 6, 7). This field contains a code which specifies the time at which the channel detected an error. This code has meaning only when CSW channel check bits 45 or 46 are set and interface bytes are present for the device. It is not to be used when an error routine is involved with a unit-check condition. The bits are interpreted as follows:

  000 - An error occurred while executing a TIO instruction. This code is used to distinguish a TIO error from a TIO clearing an error interruption.

  001 - Command Out has been sent by the channel but the unit status has not yet been analyzed by the channel. This retry code may indicate unpredictable device movement, depending on whether control units start movement on Command Out.

  010 - The command has been accepted by the device, but no data has been transferred. This code is needed for tape devices. Tape drives do not begin motion on a write command until the first byte of data has been transferred.

  011 - At least one byte of data has been transferred over the interface, and the command address in the CSW reflects this particular data transfer. In general, this means movement may have taken place if Interface Disconnect was issued.

  100 - The command has been rejected by the control unit, or Command Out has not been sent to the control unit for the command address stored in the CSW. In either case, no device movement has begun for this CCW. Hardware Test I/O may have occurred.

  101 - The command has been accepted but data transfer is unpredictable. If an Interface Disconnect is given at the time of error, most devices can be retried; a tape Write Command is an exception to this.

  110 - Spare.

  111 - No other codes apply.

CHANNEL STATUS (FOR MULTIPROCESSING) (CHANSTAT)

If the Multi-System Feature is present, CCH records the channel status for each CPU in a 2-byte field called CHANSTAT in the Inboard Record Entry, illustrated in Figure 29. The first seven bits of each byte correspond to the seven possible channels for each CPU, and each is set to one if that channel is offline. The last bit of each of the 2 bytes is unused.

EVENT CONTROL BLOCK

An event control block (ECB) is used to post the completion of an I/O request. It indicates whether or not the request has completed successfully.

The following characteristics are pertinent:

- Quantity. There is one ECB for each I/O request.
- Creation. The ECBs are created by the user of the I/O supervisor.
- Storage Areas. All ECBs reside in unprotected problem program storage.
- Size. An ECB is 4 bytes in length.
- Format. The ECB format, after it is posted complete, is shown in Figure 15.

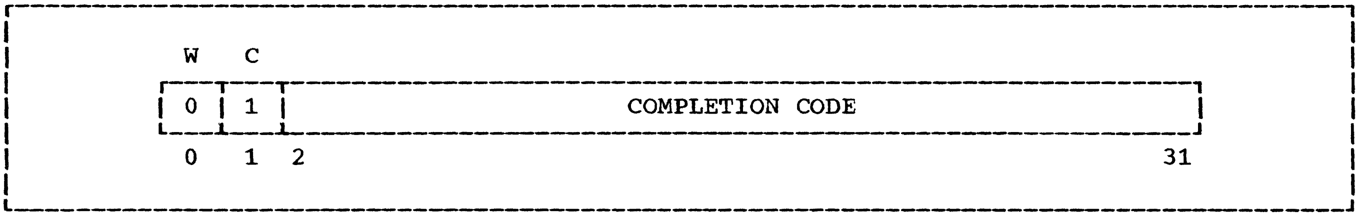


Figure 15. Event Control Block Format After Posting

Bit 0: W

This is the wait bit and is set to 0.

Bit 1: C

This is the complete bit and is always set to 1 when the I/O request has completed, regardless of the success of the event.

Bits 2-31: Completion Code (30 bits)

This code, along with the complete bit, is the ECB code as contained in the IOB. The completion code indicates to the user in what manner the I/O request has completed. When the first bit of the completion code is off, the request was not successfully completed. The next five bits indicate the cause and the remaining 24 bits are 0. There are six different completion codes; the six significant bits for these codes are:

111111 Normal completion (no errors).

000001 I/O permanent error code.

000010 Extent permanent error code. This code indicates that the seek address specified in the IOB is out of the extent specified in the DEB.

000100 IOB intercept code. Whenever an error occurs after a channel end interruption for a device, the I/O request for that device has already been posted complete and the request element returned to the freelist. To handle the error, the I/O supervisor sets the UCB intercept flag to indicate that the next I/O request for that device must be intercepted. When intercepted, the IOB for the new I/O request and the CSW and sense data for the error are passed to the error recovery procedures for the device. If a permanent error exists, the ECB for the intercepted IOB is posted complete with the IOB intercept code.

001000 Not started or purged. This code signifies either that the I/O request has not been started or that it has been purged.

001111 Error could not be retried. This code signifies that the home address and/or R0 could not be read during error recovery procedures.

INPUT/OUTPUT BLOCK

An input/output block (IOB) contains information that pertains to a requested I/O operation and is used by the I/O supervisor. The address of an IOB is passed as the only parameter of the EXCP macro instruction.

The characteristics of an IOB are as follows:

- Quantity. There must be one unique IOB for each I/O request.
- Creation. The IOBs are created by the user of the I/O supervisor.
- Storage Area. All IOBs reside in unprotected problem program storage.
- Size. An IOB is 32 or 40 bytes in length.
  1. The 32-byte field is always present for all devices.
  2. The 32-byte field plus 8 additional bytes are present for direct access and telecommunications devices.
- Format. The IOB format is shown in Figure 16.

```
     |<-------------------------4 bytes--------------------------->|
 + 0 ┌───────────────┬───────────────┬─────────────────────────────┐      -
     |   FLAGS 1     |   FLAGS 2     |        SENSE BYTES          |      ^
 + 4 ├───────────────┼───────────────┴─────────────────────────────┤      |
     |   ECB CODE    |               ECB ADDRESS                   |      |
 + 8 ├───────────────┼─────────────────────────────────────────────┤      |
     |   FLAGS 3     |                                             |      |
 +12 ├───────────────┘          CHANNEL                            |      |
     |                          STATUS WORD                        |     ALL
 +16 ├───────────────┬─────────────────────────────────────────────┤   DEVICES
     |   SIO CC      |               START ADDRESS                 |      |
 +20 ├───────────────┼─────────────────────────────────────────────┤      |
     |               |               DCB ADDRESS                   |      |
 +24 ├───────────────┼─────────────────────────────────────────────┤      |
     |REPOS. MODIFIER|               RESTART ADDRESS               |      |
 +28 ├───────────────┴───────────────┬─────────────────────────────┤      |
     |  BLK CNT INC AMT (TAPE)       |        ERROR COUNT          |      V
     └───────────────────────────────┴─────────────────────────────┘      -

 +32 ┌───────────────┬─────────────────────────────────────────────┐      -
     |   M EXTENT    |               SEEK ADDRESS                  | DIRECT ACCESS
 +36 ├───────────────┘                                             |   AND T/P
     |   (If T/P, these bytes are pointers for QTAM or BTAM.)      |
     └─────────────────────────────────────────────────────────────┘      -
```

Figure 16. Input/Output Block Format

FLAGS 1 (1 byte)
Bits 0 - 1: Channel Program Description

These two bits indicate the construction of the CCW list, and are set by the user of the I/O supervisor. They decode as follows:

00	No chaining
10	Data chaining only
01	Command chaining only
11	Data and command chaining (mixed)

Bit 2: IOB Error Correction Indicator

This bit is set by an IBM-supplied error routine to indicate that the I/O request associated with the IOB is under control of the error routine.

Bit 3: Modifier Flag 1

This bit is set by an error routine to indicate that the device in use must be repositioned by use of the command code specified by an IBM-supplied error routine in the IOB reposition modifier byte. (See Section I: Tape Start I/O Module.)

Bit 4: Modifier Flag 2

This bit is set by an IBM-supplied error routine to indicate that cyclic redundancy check (CRC) correction is required for the tape. (See Section I: Tape Start I/O Module.)

Bit 5: IOB Exception

This bit is set to indicate that an error occurred during the I/O request associated with this IOB. The bit is turned off when the error is corrected; if uncorrected, the exception bit is left on.

Bit 6: Unrelated

This bit is set by the user of the I/O supervisor and identifies all I/O requests (for use of a data set) that are not related to any other request. Requests that are related are executed in a first-in first-out (FIFO) arrangement and no related request is started until all prior related requests have completed. (Related is 0, unrelated is 1.)

Bit 7: Start/Restart

This bit is set by an IBM-supplied error routine to indicate that the restart address in the IOB is to be used for the I/O request; when 0, the start address is used.

FLAGS 2 (1 byte)

Bit 0: Halt I/O Issued Flag

This bit is set by the SVC purge routine to indicate that it has issued a halt input/output (HIO) instruction for the device in use for the I/O request. The request is associated with the IOB and is being purged.

Bit 1: Sense Flag

This bit indicates that a sense command should be issued by the sense subroutine because of an error at the previous channel end. At device end, the device is free and the sense command is issued.

Bit 2: Purge Flag

This bit is turned on by the SVC or I/O purge routine when awaiting completion of the device and allowing the I/O request to quiesce.

Bit 3: Read HA R0 Flag

This bit is turned on by the direct access error routines when no seek is required.

Bit 4: No Test for Out-of-Extent Flag

This bit is turned on by the direct access error routines when an alternate track is in use.

Bit 5: CCHH Update Flag

This bit is turned on by the direct access device error routines when updating the seek address because of a cylinder end or file mask violation condition.

Bit 6: Device End Post Flag

This bit is turned on by the Device End Post trapcode module to signify that a graphic class device has had Device End status ORed with Channel End Status.

Bit 7: IOB QSAM Flag

This bit is turned on by the Queued Sequential Access Method routines when error recovery (using three buffers) is to be provided for a 2540 card punch.

SENSE BYTES (2 bytes)
The sense subroutine loads these bytes with the first 2 bytes of sense information read into the UCB when a sense command is given.

ECB CODE (1 byte)
The I/O supervisor places the completion code for the I/O request associated with the IOB into this area. This code is also placed into the event control block (ECB) when the request is posted complete. (For additional information, see "Event Control Block".)

ECB ADDRESS (3 bytes)
These bytes are set by the user of the I/O supervisor to contain the address of the event control block (ECB) that corresponds to the IOB and is to be posted complete at the completion of the I/O request described in the IOB.

FLAGS 3 (1 byte)
This byte is used by IBM-supplied error routines and is defined in Section III: Device-Dependent Characteristics.

CHANNEL STATUS WORD (7 bytes)
These bytes contain the low-order seven bytes of the channel status word (CSW), as stored by the I/O supervisor at channel end.

SIO CC (1 byte)
Bits 2 and 3 (the third and fourth bits) of this byte are set by the start I/O subroutine to contain the condition code resulting from issuance of the SIO instruction for the associated I/O request.

START ADDRESS (3 bytes)
These three bytes are supplied by the user of the I/O supervisor and contain the address of the first CCW that is to be executed in the channel program.

DCB ADDRESS (3 bytes)
These bytes are supplied by the user of the I/O supervisor to contain the address of the data control block (DCB) that corresponds to the IOB.

REPOSITION MODIFIER (1 byte)
This byte is loaded with a positioning command code by IBM-supplied error routines that are used for device repositioning procedures when the IOB modifier flag 1 is on.

RESTART ADDRESS (3 bytes)
This address is used as the CCW address in the channel program for an error restart by IBM-supplied error routines during error correction procedures.

BLOCK COUNT INCREMENT AMOUNT (2 bytes)
This is the tape block count increment amount set by the user of the I/O supervisor and is used by the I/O supervisor at device end (completion of the I/O request) to update the tape block count (BLKCT) field in the DCB associated with the IOB. The field is signed and may contain an increment or decrement up to ±32767. When an uncorrectable error occurs, the I/O supervisor uses this field to update the DCB block count as usual.

ERROR COUNT (2 bytes)
These bytes contain flags and counts used by the IBM-supplied error routines and are explained in Section III: Device-Dependent Characteristics.

M EXTENT (1 byte)
This byte is set by the user of the I/O supervisor to indicate the number of the DEB extent that is to be used for the request. (The first extent is indicated by the character 0, the second by 1, the third by 2, etc.)

SEEK ADDRESS (7 bytes)
The user supplies the initial seek address BBCCHHR required for the I/O request for the direct access device. For teleprocessing devices, this field contains QTAM and BTAM pointers.

LOGICAL CHANNEL WORD TABLE

The logical channel word table consists of the logical channel words that control the logical channel queues. It is used by the I/O supervisor and the I/O purge and SVC purge routines.

The logical channel word table has the following characteristics:

- Creation. The table is created at system generation time.
- Storage Area. It resides, as a permanent part of the resident supervisor, in protected resident storage (when protection is available).
- Size. The table contains one 8-byte logical channel word per logical channel queue.
- Means of Access. The LCHTAB byte in the UCB, when multiplied by 8 and added to the starting address of the logical channel word table, gives the proper logical channel word for the device and its associated logical channel queue.
- Format. The format of a logical channel word is shown in Figure 17.

FIRST REQUEST	LAST REQUEST	SCRATCH	TCH MOD ADDR
<---2 bytes--->	<---2 bytes --->	<---2 bytes --->	<---2 bytes --->
<----------------------------8 bytes--------------------------------->			

Figure 17. Logical Channel Word Format

FIRST REQUEST (2 bytes)
These two bytes contain either an address or an index value to the first request element in the logical channel queue.

LAST REQUEST (2 bytes)
These two bytes contain either an address or an index value to the last request element in the logical channel queue.

Notes:

1. When a logical channel queue is void, the "first request" field contains a dummy link address of hexadecimal FFFF and the "last request" field contains the address of that logical channel word.
2. When there is only one request element in the queue, both "first request" and "last request" contain the address of that element.

SCRATCH (2 bytes)
This field is used as a temporary storage area for an address or index value. It is used when more than one logical channel queue for a physical channel is searched in order to find the highest priority I/O request with which to restart the channel.

TCH MOD ADDR (2 bytes)
This field addresses the device-dependent test channel module.

REQUEST ELEMENT TABLE

The elements in the request element table are used by the I/O supervisor to represent active or queued I/O requests. The unused elements in the table are available for incoming I/O request representation.

The request element table has the following characteristics:

- Creation. The table is created at system generation time.

- Storage Area. It resides, as a permanent part of the resident supervisor, in protected resident storage (when protection is available).

- Size. The total number of request elements in the table is defined at system generation time. The request element table for a system in which MVT is excluded contains a 12-byte request element for the maximum number of I/O requests expected at any one time; and for a system in which MVT is included, a 16-byte request element.

- Means of Access. The active request elements are addressed by the LAST REQUEST field in the associated UCB. The available request elements are contained in the freelist, which is addressed by the freelist pointer in the CVT. The queued request elements are within the particular logical channel queue referred to by the logical channel word.

- Format. The I/O supervisor is concerned with all information in a request element. The format of a 12-byte and 16-byte request element is shown in Figure 18.

Format of a 12-byte Request Element

LINK FIELD		UCB ADDRESS
TASK ID	IOB ADDRESS	
PRIORITY	DEB ADDRESS	
<1 byte>	<1 byte>	<----2 bytes---->

Format of a 16-Byte Request Element

LINK FIELD		UCB ADDRESS
--	IOB ADDRESS	
PRIORITY	DEB ADDRESS	
KEY	TCB ADDRESS	
<1 byte>	<1 byte>	<----2 bytes---->

Figure 18. Request Element Formats

LINK FIELD (2 bytes)
This is a 2-byte link field used to link the request elements that are members of a particular queue or belong to the freelist.

UCB ADDRESS (2 bytes)
This field addresses the UCB associated with the queued I/O request.

TASK ID (1 byte)
This byte contains the task control block identification of the task that originally initiated the I/O request.

IOB ADDRESS (3 bytes)
This field contains the address of the IOB associated with the I/O request.

PRIORITY (1 byte)
This byte contains the priority of the I/O request represented by this request element. The priority is assigned at open time according to the priority of the associated task.

DEB ADDRESS (3 bytes)
This field contains the address of the DEB associated with the data set for this I/O request.

KEY (1 byte)
This field contains the protect key associated with the request.

TCB Address (3 bytes)
This field contains the address of the task control block for the task that initiated the I/O request.

STATISTICS TABLE

The statistics table contains the statistical data required to maintain statistical data records. It is used by IBM-supplied error routines and the statistical data recorder (SDR) of SER.

The statistics table has the following characteristics:

- Creation. The statistics table is created at system generation time.
- Storage Area. The table resides, as a permanent part of the resident supervisor, in protected resident storage (when protection is available).
- Size. The statistics table contains from 0 to 128 ten-byte entries; there is one entry for each device attached to the system. There are however, 9 entries for a 2314.
- Means of Access. The STATAB byte in the UCB contains a factor, which when multiplied by ten and added to the starting address of the statistics table, provides the address of the proper entry for a device. For the 2314, the low-order four bits of the fifth sense byte are added to STATAB to get the correct entry.
- Format. The format of each entry in the table varies with the type of device. The first 8 bytes of an entry contain statistical data; the last 2 bytes are a work area used by error routines. The device type formats are shown in Figure 19; note that the work area is not shown.

1052
2150
1015
1285
1402
1442
1404
2201
1403
1443
2501
2520
2671
2701
2702
7770
7772
2250
2260
1053
2280
2282

Unit Record Equipment

Temporary Read Failures	Temporary Write Failures
	Bus Out Check
Equipment Check	Overrun
Device-Dependent	Device-Dependent

|<------------1 byte------------->|

2400 Tape Series

Temporary Read Failures	Temporary Write Failures
Intervention Required	Bus Out Check
Equipment Check	Overrun
Word Count 0	Data Converter Check
R/W Vertical Redundancy Check	Longitudinal Redundancy Check
Skew	Cyclic Redundancy Check
Skew Reg VRC	Noise

|<------------1 byte------------->|

Figure 19. Statistics Table Entry Formats (Part 1 of 2)

2841 Control Unit

Temporary Read Failures	Temporary Write Failures
Intervention Required	Bus Out Check
Equipment Check	Overrun
Track Condition	Seek Check
Unsafe	
Serializer/ Deserializer	Control Unit Tag Line
Arithmetic Logical Unit	
Missing Address Marker	

|<------------1 byte-------------->|

2820 Control Unit

Temporary Read Failures	Temporary Write Failures
Intervention Required	Bus Out Check
Equipment Check	
Track Condition Check	
	Track Overrun
No Record Found	

|<------------1 byte-------------->|

2280

Temporary Read Failures	Temporary Write Failures
Intervention Required	Bus Out Check
Equipment Check	
Byte Count 0	
Recorder Forced Gap	
	Graphics
Work area	
Work area	

|<------------1 byte-------------->|

• Figure 19. Statistics Table Entry Formats (Part 2 of 2)

UCB LOOKUP TABLE

The UCB lookup table is used by the I/O interruption supervisor to obtain the address of the UCB associated with an I/O interruption.

The UCB lookup table has the following characteristics:

- Creation. The table is created at system generation time.
- Storage Area. The table resides, as a permanent part of tne resident supervisor, in protected resident storage (when protection is available).
- Size. The size of the table is dependent upon the number and the unit addresses of I/O devices, control units, and physical channels attached to the system.
- Means of access. The table values are used in the algorithm routine described in Section II: UCB Lookup Routine. The table is addressed by the CVT. The algorithm shown in Figure 20 is used to obtain the address of the UCB.
- Format. The UCB lookup table is segmented as shown in Figure 21.

IECILK1 + 3-bit channel address -----> K
IECILK1 + 4-bit control unit address + K -----> L
IECILK2 + 2 (4-bit device address) + 2L -----> the actual UCB address
THE -----> reads "is the address of."

IECILK1
is the starting address of the UCB lookup table.

IECILK2
is the starting address of the UCB address list portion of the UCB lookup table.

Note: The addresses of both the IECILK1 and the IECILK2 are contained in the CVT.

K
is the index value obtained from the channel portion of the UCB lookup table.

L
is the index value obtained from the control unit portion of the UCB lookup table.

Actual UCB address
is the 2-byte address of the UCB associated with the I/O interruption. It is obtained from the UCB address list portion of the UCB lookup table.

Figure 20. UCB Lookup Algorithm

K_1	K_2	K_3	K*	K*	K*	K*	L_1	L_2	L_3	$L_{n\ 2}$	$L_{n\ 1}$	L_n	UCB01	UCB02	UCBN
<-----------Channel Portion---------->							<-----Control Unit Portion------>						<UCB Address List>		

K_n (1 byte)
The channel portion contains index values that are relative to the starting address of the entire UCB lookup table.

L_n (1 byte)
The control unit portion contains index values that are relative to the starting address of the UCB address list.

UCB_n (2 bytes)
The UCB address list contains the addresses of the UCBs in the system.

Figure 21. UCB Lookup Table Format

UNIT CONTROL BLOCK

Every device is represented by a unit control block (UCB). A UCB indicates device activity and provides the linkage to device-dependent code and tables. The properties of a UCB follow:

- Creation. The UCBs are created at system generation time.

- Storage Area. All UCBs are located below 32K in protected resident storage (when protection is available), and are a permanent part of the resident supervisor.

- Size. A UCB is from 24 to 288 bytes.

- Quantity. There is one UCB per unique, physical device address defined for the system. Each telecommunications line on a transmission control unit and a device attached to a shared control unit or a tape unit switch has only one UCB. A 2314 direct access storage facility has eight UCBs.

- Format. The UCB format is shown in Figure 22. (For the format of a UCB for a 2321 data cell, refer to the publication, IBM System/360 Operating System: System Control Blocks.)

```
Hex  Dec  |<-------------------------------4 bytes------------------------------->|
- 4  - 4  +-----------------+-----------------------------------------------------+      -
          |   M65MP FLAGS   |                       NOT USED                      |  M65MP ONLY
          +-----------------+-----------------------------------------------------+      -

+ 0  + 0  +-----------------+-----------------+-----------------+-----------------+      -
          |INTERNAL JOB NO  |ALLOC CHAN MASK  |  UCB ID        X|    STATUS  A    |      ^
+ 4  + 4  +-----------------+-----------------+-----------------+-----------------+      |
          |    CHAN ADDR    |UNIT ADDRESS    X|  FLAGS1         |DEVTAB INDEX    X|      |
+ 8  + 8  +-----------------+-----------------+-----------------+-----------------+      |
          |    ERRTAB      X|STATAB INDEX    X|LCHTAB INDEX    X|    ATNTAB      X|     ALL
+ C  +12  +-----------------+-----------------+-----------------+-----------------+   DEVICES
          |FLAGS/CHAN MASK  |                 UNIT  NAME                         X|      |
+10  +16  +--------+--------+-----------------------------------------------------+      |
          |        |                       DEVICE TYPE                           X|      |
+14  +20  +--------+--------------------------+-----------------------------------+      |
          |    LAST REQUEST ELEMENT           |        SENSE INFORMATION          |      v
+18  +24  +-----------------------------------+-----------------------------------+      -

+18  +24  +-----------------------------------------------------------------------+      -
          |                               UCBUCSID                                |     UCS
+1C  +28  +-----------------------------------------------------------------------+   PRINTER
          |                               UCBUCSOP                                |  EXTENSION
+20  +32  +-----------------------------------------------------------------------+      -

+18  +24  +-----------------------------------+-----------------+-----------------+      -
          |         SENSE INFORMATION         |    USE COUNT    |  CONTROL BYTE   |      ^
+1C  +28  +-----------------------------------+-----------------+-----------------+      |
          |                          TASK ENTRY ADDRESS                           |      |
+20  +32  +-----------------------------------------------------------------------+   GRAPHIC
          |                           RESTART ADDRESS                             |   DEVICES
+24  +36  +-----------------+-----------------------------------------------------+      |
          |  DEVICE INDEX   |                BUFFER TABLE ADDRESS                 |      |
+28  +40  +-----------------+-----------------------------------------------------+      -

+18  +24  +-----------------------------------------------------------------------+      -
          |                          SENSE INFORMATION                            |      ^
+1C  +28  +-----------------------------------------------------------------------+      |
          |                        VOLUME SERIAL NUMBER                           |      |
+20  +32  |                                   +-----------------+-----------------+      |
          |                                   |    STATUS B     | DATA MGMT CTR   |   MAGNETIC
+24  +36  +-----------------------------------+-----------------+-----------------+     TAPE
          |          SEQUENCE COUNT           |        SEQUENCE NUMBER            |   DEVICES
+28  +40  +-----------------------------------+-----------------------------------+      |
          |                       DATA SET SERIAL NUMBER                          |      |
          |                                   +-----------------------------------+      |
          |                                   |              RESERVED             |      |
+30  +48  +-----------------------------------+-----------------------------------+      |
          |                    CCW FOR READ-OPPOSITE RECOVERY                     |      v
+37  +55  +-----------------------------------------------------------------------+      -
```

• Figure 22. Unit Control Block Format (Part 1 of 2)

```
Hex Dec  |<------------------------4 bytes------------------------->|
+18 +24  +-----------------------------------------------------------+      -
         |                   SENSE INFORMATION                       |      ^
+1C +28  +-----------------------------------------------------------+      |
         |                  VOLUME SERIAL NUMBER                     |      |
+20 +32  |                              +------------+---------------+      |
         |                              |  STATUS B  | DATA MGMT CTR |      |
+24 +36  +------------------------------+------------+---------------+      |
         |                     ADDRESS OF VTOC                       |      |
+28 +40  +----------------+-----------+------------------------------+      |
         |Reserve Count   |  Note 2   |                              |      |
         +----------------+ +29,+41---+    RESERVED FOR IOS          |      |
         |                                                           |      |
+30 +48  +-----------------------------------------------------------+      |
         |                                                           |   DIRECT
         |         DIRECT ACCESS ADDRESS OF LAST SEEK                |   ACCESS
         |                                                           |   DEVICES
+38 +56  +----------------+------------------------------------------+      |
         |NUMBER OF USERS |      DIRECT ACCESS ECB ADDRESS           |      |
+3C +60  +----------------+------------------------------------------+      |
         |                                                           |      |
         |               ERROR RECOVERY WORK AREA                    |      |
         |                                                           |      |
+64 +100 +-----------------------------------------------------------+      |
         |                                                           |      |
         |                  OVERFLOW WORK AREA                       |      |
         |                                                           |      v
+8C +140 +-----------------------------------------------------------+      -

        Note 1:  The X areas are loaded at system generation time; all
                 except ATNTAB, which is changed by system routines,
                 remain constant.

        Note 2:  OUTSTANDING RESERVE INDICATOR
```

•Figure 22. Unit Control Block Format (Part 2 of 2)

M65MP FLAGS (1 byte)

Bits 0-1: Not used.

Bit 2: Set to 1 by CPUA if CPUB is to perform a Halt I/O operation following a shoulder tap.

Bit 3: Set to 1 by CPUB if CPUA is to perform a Halt I/O operation following a shoulder tap.

Bit 4: Set to 0 if CPUA started the last I/O operation; set to 1 if CPUB started the last I/O operation.

Bit 5: Not used.

Bit 6: Set to 1 by the Nucleus Initialization Program (NIP) if this device is not accessible from CPUB.

Bit 7: Set to 1 by the Nucleus Initialization Program (NIP) if this device is not accessible from CPUA.

UCB ID (1 byte)

This is an encoded field of hexadecimal "FF" that identifies this block as a UCB.

HIO (1 bit)

This bit is turned on when a halt I/O (HIO) instruction is issued for the device represented by this UCB.

SM (1 bit)

This bit indicates that a sense command was attempted for the represented device, but that the CSW status modifier flag was set in response to the sense command. When the device is free, the sense is reissued.

Bits 2, 3, and 4: Not Used.

CHANNEL ADDRESS (3 bits)

Bits 5 to 7 contain the physical channel address of the last physical channel used for the device represented by this UCB.

UNIT ADDRESS (1 byte)
This byte gives the absolute device and control unit address that pertains to the device represented by this UCB. The address of the physical channel to be used by the device is calculated when a physical channel becomes available. The total I/O address, consisting of the unit address and the channel address, is used in the start I/O (SIO) instruction.
Note: A device accessed through a tape unit switch must always have the same control unit address.

FLAGS 1

Bit 0: UCB Busy
This bit is turned on by the I/O supervisor when the represented device is started and remains on until device end.

Bit 1: UCB Not Ready
This bit is turned on when the I/O supervisor determines a device not-ready condition and indicates that operator intervention is required. A message is displayed to the operator by the error routine handling the condition.

Bit 2: Post
This bit indicates that channel end is awaited.

Bit 3: UCB Intercept
This bit indicates an error has occurred after channel end for the last request processed on the device and that an error routine must be called when the next request for the device is given. (Refer to IOB intercept code in "Event Control Block".)

Bit 4: Control Unit Busy
This bit is turned on whenever a bit setting of busy and status modifier are detected in the CSW after issuance of a SIO instruction. The setting of the control unit busy bit indicates that requests issued for the represented device by means of the EXCP macro instruction cannot be started. This bit does not, however, prevent queued requests for the device from being started by the channel restart subroutine. This bit is necessary because the control unit end CSW indication sometimes presents itself to the channel with a different address than was presented with control unit busy.

Bit 5: Disk Data Transfer
This bit indicates, when the UCB represents a direct access device, that the device is currently involved in a data transfer operation.

Bit 6: Disk Arm Seeking
This bit indicates, when the UCB represents a direct access device, that the device is currently involved in a seeking operation.

Bit 7: Error Routine in Control
This bit is turned on by an error routine to indicate that error recovery procedures are in progress for the device and that any related requests must not be started.

DEVTAB (1 byte)
This field provides the indexing value (in bytes) from the beginning of the device table to the significant device class entry in the table for the device represented by the UCB. The device table entry contains the addresses of the device-dependent enqueue, start I/O, and trapcode modules for the device type. (For additional information, refer to "Device Table.")

ERRTAB (1 byte)
This byte contains a numeric constant that is used by the exit effector to complete the 8-byte name of an IBM-supplied error routine within the SVC library for the represented device.

STATAB (1 byte)
At SYSGEN time, a device number ranging from 0 to 128 is assigned to each I/O device attached to the system. The number for the represented device is stored in this byte. The statistics table contains an 8-byte data area, in which information regarding the device's operation is maintained, and a 2-byte work area for each device. The I/O supervisor locates each data area by using the product of the STATAB byte and ten as an indexing value. This value is constant for the eight 2314 UCBs. (Refer to "Statistics Table.")

LCHTAB (1 byte)
This field, when multiplied by eight, gives the indexing value (in bytes) from the beginning of the logical channel word table to that logical channel word associated with the represented device. The logical channel word indicates the status of the particular logical channel queue to be used when the I/O supervisor must queue a request for the device. (For additional information, refer to "Logical Channel Word Table.")

ATNTAB (1 byte)
This field contains the indexing value (in bytes), set by system routines, from the beginning of the attention

table to one of its 4-byte entries. The entry contains the actual address of the attention routine required for the represented device. (For additional information, refer to "Attention Table.")

FLAGS AND CHANNEL MASK (1 byte)

Bits	Bit Settings	Meaning
0	1	SYSIN
1	1	SYSOUT
2	1	This device will be allocated for a public volume request.
3		Reserved
4-7	(variable)	I/O supervisor channel mask

UNIT NAME (3 bytes)

This field contains a symbolic name in EBCDIC characters for the device represented by the UCB. This name is used for device allocation and also by error routines in messages to the operator concerning this device.

DEVICE TYPE (4 bytes)

The contents of this field describe the type of device and pertinent characteristics of it. The first 4 bits of the first byte are flags that decode as follows:

Bit 0: Unassigned.

Bit 1: Overrunable Device
When on, this bit indicates that the device is overrunable.

Bit 2: Burst Device
When on, it indicates that the device operates in burst mode; when off, the device operates in byte mode.

Bit 3: Data Chaining
When this bit is on data chaining is allowed for the device. If it is off, data chaining is not allowed.

The remaining bits and bytes are not used by the I/O supervisor; they are discussed in the publication IBM System/360 Operating System: System Control Blocks.

LAST REQUEST (2 bytes)

The I/O supervisor issues a request element to represent each I/O request for a device. When the I/O request is started, the address of the request element is placed into this field (LAST REQUEST) which then indicates the last I/O request for the device. The information in the request element is used during interruption processing. (For additional information, refer to "Request Element Table.")

SENSE (2 to 6 bytes)

When the I/O supervisor issues a sense command, up to 6 bytes of sense information from the device's control unit is read into this area of the UCB. The number of bytes of sense information that is read varies with the device type; however, all information significant to the use of the device is normally provided within the first two bytes. The I/O supervisor then places these two bytes into the IOB.

RESERVE COUNT (for Shared DASD option only)

This area contains the reserve count for shared direct access devices. The count is incremented by a RESERVE macro instruction. If the reserve count is greater than zero, the device is reserved for a particular CPU; other CPUs may not use it. The reserve count is decremented by a DEQ macro instruction.

OUTSTANDING RESERVE INDICATOR (1 byte)

This field is used only for the Shared DASD option. The field is set by the direct access SIO module when a Start I/O instruction for a data transfer has been successful. The module moves the contents of the Reserve Count field (hex. +28) to the Outstanding Reserve indicator. The field is tested by the supervisor's ENQ/DEQ Purge routine (IEAQEQ01) during an abnormal termination. If the field is nonzero, the ENQ/DEQ Purge routine concludes that the device has been physically reserved by the terminating task's CPU and must be released.

LAST SEEK ADDRESS (8 bytes)

This area contains the seek address MBBCCHHR for the last seek on the direct access device represented by the UCB.

CCW FOR READ-OPPOSITE RECOVERY (8 bytes)

This area is used by IBM-supplied magnetic tape error routines to build a CCW for opposite-direction recovery attempts. Such recovery is tried after a data check during a read or read-backwards operation.

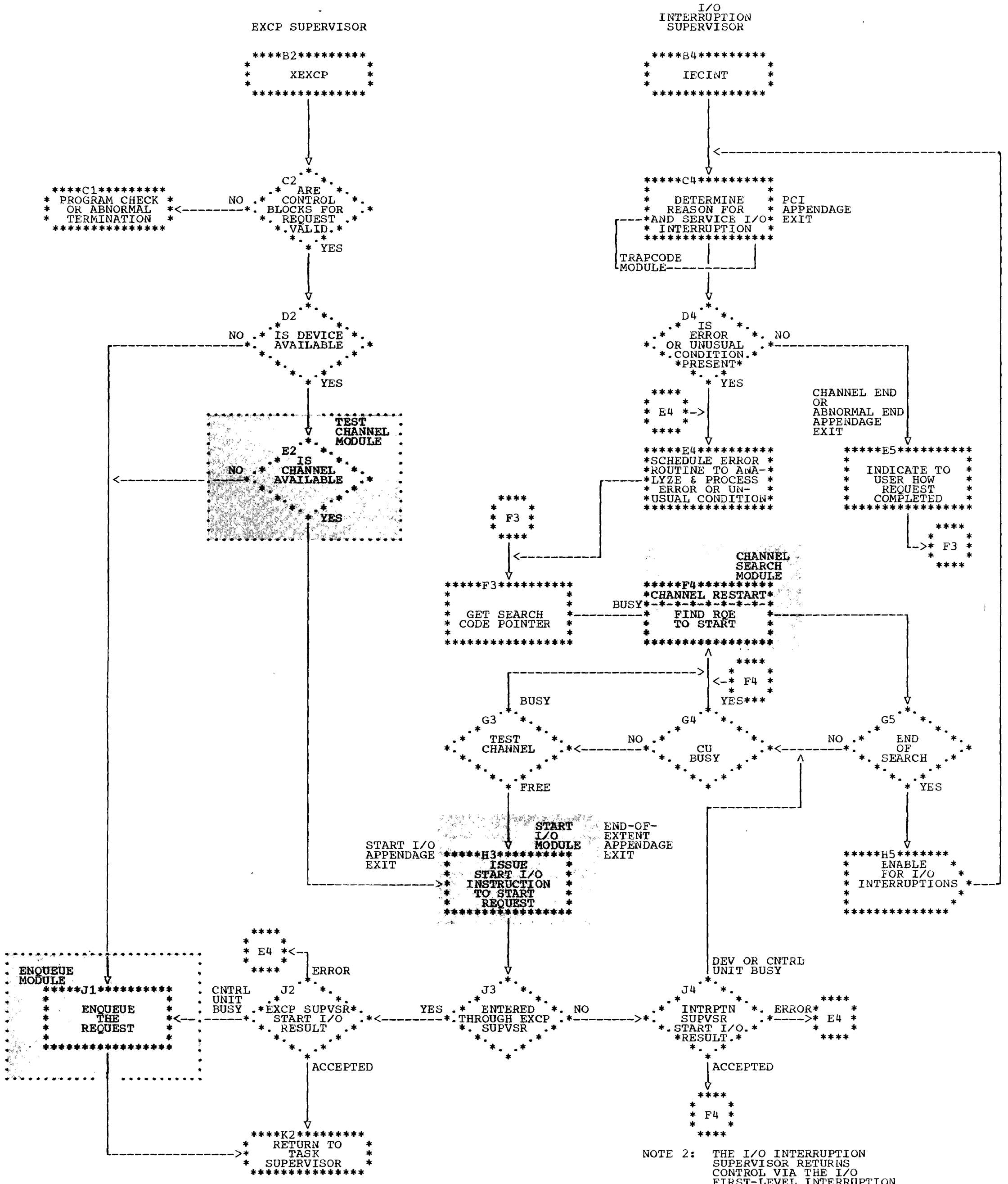
ERROR RECOVERY WORK AREA (40 bytes)

This area is used by IBM-supplied error routines for the CCWs necessary for the alternate track procedures and the head and cylinder switching procedures for direct access devices.

OVERFLOW WORK AREA (40 bytes)

This area is used if the track overflow feature is installed.

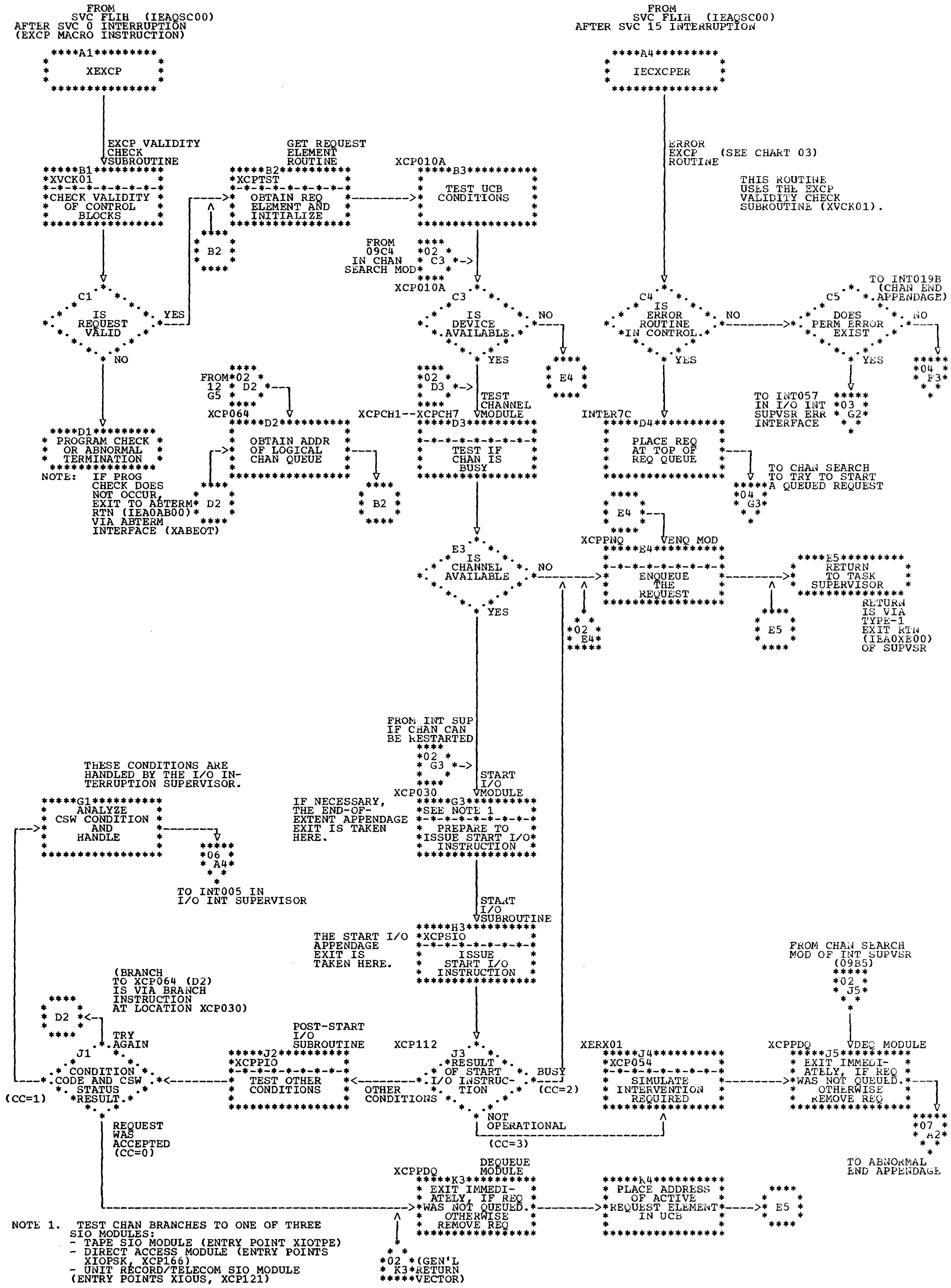
• Chart 01. Functional Flow of the Input/Output Supervisor


EXCP SUPERVISOR
I/O INTERRUPTION SUPERVISOR
B2 XEXCP
B4 IECINT
C1 PROGRAM CHECK OR ABNORMAL TERMINATION
NO
C2 ARE CONTROL BLOCKS FOR REQUEST VALID.
YES
C4 DETERMINE REASON FOR AND SERVICE I/O INTERRUPTION
PCI APPENDAGE EXIT
TRAPCODE MODULE
D2 IS DEVICE AVAILABLE
NO
YES
D4 IS ERROR OR UNUSUAL CONDITION PRESENT
NO
YES
E4
TEST CHANNEL MODULE
E2 IS CHANNEL AVAILABLE
NO
YES
CHANNEL END OR ABNORMAL END APPENDAGE EXIT
E4 SCHEDULE ERROR ROUTINE TO ANALYZE & PROCESS ERROR OR UNUSUAL CONDITION
E5 INDICATE TO USER HOW REQUEST COMPLETED
F3
F3
CHANNEL SEARCH MODULE
F3 GET SEARCH CODE POINTER
BUSY
F4 CHANNEL RESTART FIND RQE TO START
F4
YES
BUSY
G3 TEST CHANNEL
NO
G4 CU BUSY
NO
G5 END OF SEARCH
FREE
YES
START I/O MODULE
END-OF-EXTENT APPENDAGE EXIT
START I/O APPENDAGE EXIT
H3 ISSUE START I/O INSTRUCTION TO START REQUEST
H5 ENABLE FOR I/O INTERRUPTIONS
E4
ERROR
ENQUEUE MODULE
J1 ENQUEUE THE REQUEST
CNTRL UNIT BUSY
J2 EXCP SUPVSR START I/O RESULT
YES
J3 ENTERED THROUGH EXCP SUPVSR
NO
DEV OR CNTRL UNIT BUSY
J4 INTRPTN SUPVSR START I/O RESULT
ERROR
E4
ACCEPTED
ACCEPTED
F4
K2 RETURN TO TASK SUPERVISOR

NOTE 1: THE EXCP SUPERVISOR RETURNS CONTROL VIA THE TYPE-1 EXIT RTN (IEA0XE00) OF THE SUPERVISOR.

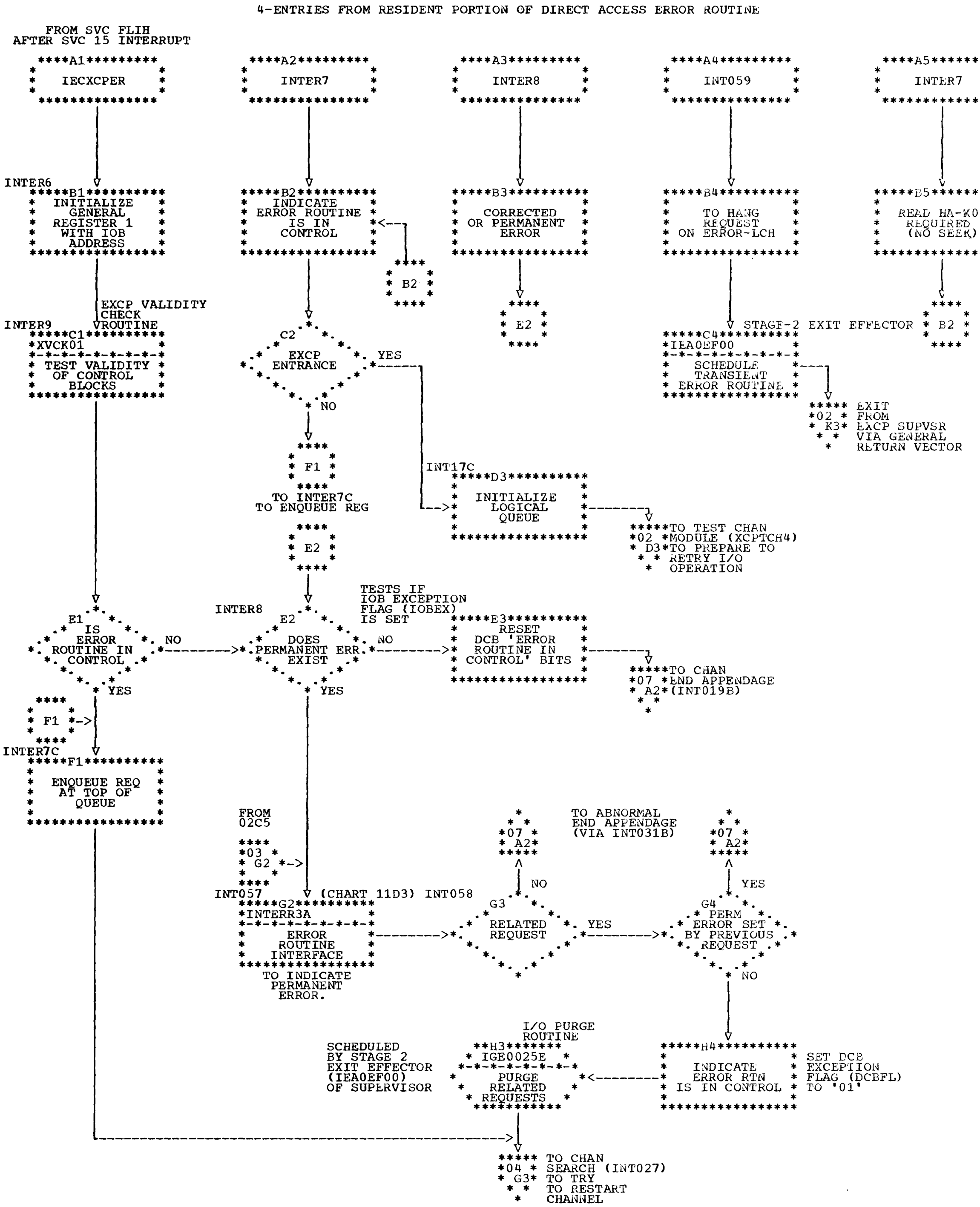
NOTE 2: THE I/O INTERRUPTION SUPERVISOR RETURNS CONTROL VIA THE I/O FIRST-LEVEL INTERRUPTION HANDLER (IEAQIO00) AND THE DISPATCHER (IEAODS) OF THE SUPERVISOR.

• Chart 02. Control Flow Within the EXCP Supervisor

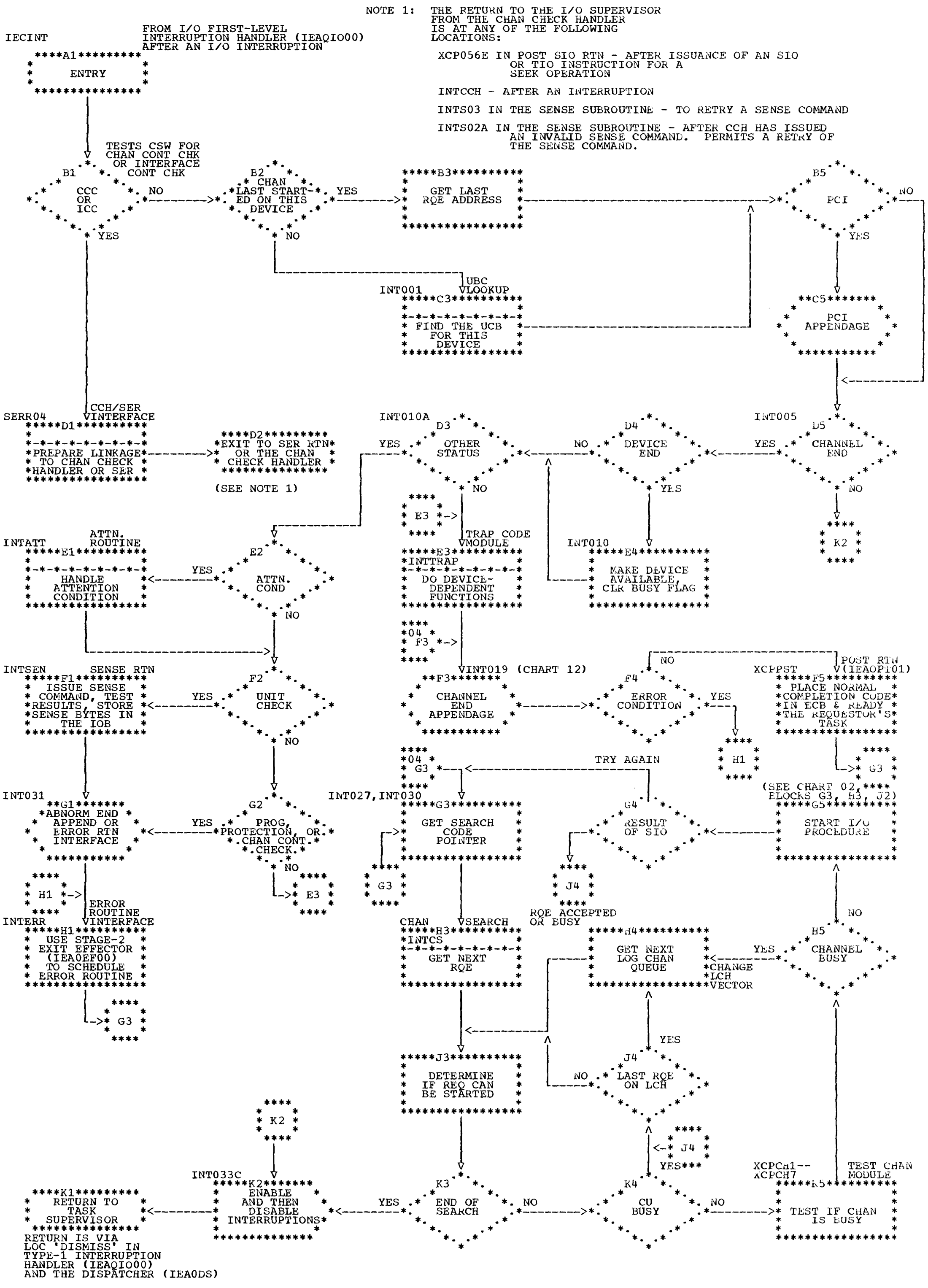

FROM SVC FLIH (IEAQSC00) AFTER SVC 0 INTERRUPTION (EXCP MACRO INSTRUCTION)
A1
XEXCP
EXCP VALIDITY CHECK SUBROUTINE
B1
XVCK01
CHECK VALIDITY OF CONTROL BLOCKS
C1
IS REQUEST VALID
YES
NO
D1
PROGRAM CHECK OR ABNORMAL TERMINATION
NOTE: IF PROG CHECK DOES NOT OCCUR, EXIT TO ABTERM RTN (IEA0AB00) VIA ABTERM INTERFACE (XABEOT)
GET REQUEST ELEMENT ROUTINE
B2
XCPTST
OBTAIN REQ ELEMENT AND INITIALIZE
B2
FROM 12 G5
02 D2
XCP064
D2
OBTAIN ADDR OF LOGICAL CHAN QUEUE
D2
B2
XCP010A
B3
TEST UCB CONDITIONS
FROM 09C4 IN CHAN SEARCH MOD
02 C3
XCP010A
C3
IS DEVICE AVAILABLE
NO
YES
E4
02 D3
TEST CHANNEL MODULE
XCPCH1--XCPCH7
D3
TEST IF CHAN IS BUSY
E3
IS CHANNEL AVAILABLE
NO
YES
FROM SVC FLIH (IEAQSC00) AFTER SVC 15 INTERRUPTION
A4
IECXCPER
ERROR EXCP ROUTINE
(SEE CHART 03)
THIS ROUTINE USES THE EXCP VALIDITY CHECK SUBROUTINE (XVCK01).
C4
IS ERROR ROUTINE IN CONTROL
NO
YES
C5
DOES PERM ERROR EXIST
TO INT019B (CHAN END APPENDAGE)
NO
04 F3
YES
TO INT057 IN I/O INT SUPVSR ERR INTERFACE
03 G2
INTER7C
D4
PLACE REQ AT TOP OF REQ QUEUE
TO CHAN SEARCH TO TRY TO START A QUEUED REQUEST
04 G3
E4
XCPPNQ
ENQ MOD
E4
ENQUEUE THE REQUEST
02 E4
E5
RETURN TO TASK SUPERVISOR
E5
RETURN IS VIA TYPE-1 EXIT RTN (IEA0XE00) OF SUPVSR
FROM INT SUP IF CHAN CAN BE RESTARTED
02 G3
START I/O MODULE
XCP030
G3
SEE NOTE 1
PREPARE TO ISSUE START I/O INSTRUCTION
IF NECESSARY, THE END-OF-EXTENT APPENDAGE EXIT IS TAKEN HERE.
THESE CONDITIONS ARE HANDLED BY THE I/O INTERRUPTION SUPERVISOR.
G1
ANALYZE CSW CONDITION AND HANDLE
06 A4
TO INT005 IN I/O INT SUPERVISOR
START I/O SUBROUTINE
H3
XCPSIO
ISSUE START I/O INSTRUCTION
THE START I/O APPENDAGE EXIT IS TAKEN HERE.
(BRANCH TO XCP064 (D2) IS VIA BRANCH INSTRUCTION AT LOCATION XCP030)
D2
TRY AGAIN
J1
CONDITION CODE AND CSW STATUS RESULT
(CC=1)
REQUEST WAS ACCEPTED (CC=0)
POST-START I/O SUBROUTINE
J2
XCPPIO
TEST OTHER CONDITIONS
OTHER CONDITIONS
XCP112
J3
RESULT OF START I/O INSTRUCTION
BUSY
(CC=2)
NOT OPERATIONAL
(CC=3)
XERX01
J4
XCP054
SIMULATE INTERVENTION REQUIRED
FROM CHAN SEARCH MOD OF INT SUPVSR (09B5)
02 J5
XCPPDQ
DEQ MODULE
J5
EXIT IMMEDIATELY, IF REQ WAS NOT QUEUED. OTHERWISE REMOVE REQ
07 A2
TO ABNORMAL END APPENDAGE
XCPPDQ
DEQUEUE MODULE
K3
EXIT IMMEDIATELY, IF REQ WAS NOT QUEUED. OTHERWISE REMOVE REQ
02 K3 (GEN'L RETURN VECTOR)
K4
PLACE ADDRESS OF ACTIVE REQUEST ELEMENT IN UCB
E5

NOTE 1. TEST CHAN BRANCHES TO ONE OF THREE SIO MODULES:
- TAPE SIO MODULE (ENTRY POINT XIOTPE)
- DIRECT ACCESS MODULE (ENTRY POINTS XIOPSK, XCP166)
- UNIT RECORD/TELECOM SIO MODULE (ENTRY POINTS XIOUS, XCP121)

• Chart 03. Control Flow Within the Error EXCP Routine


4-ENTRIES FROM RESIDENT PORTION OF DIRECT ACCESS ERROR ROUTINE
FROM SVC FLIH AFTER SVC 15 INTERRUPT
A1 IECXCPER
A2 INTER7
A3 INTER8
A4 INT059
A5 INTER7
INTER6
B1 INITIALIZE GENERAL REGISTER 1 WITH IOB ADDRESS
B2 INDICATE ERROR ROUTINE IS IN CONTROL
B2
B3 CORRECTED OR PERMANENT ERROR
E2
B4 TO HANG REQUEST ON ERROR-LCH
B5 READ HA-R0 REQUIRED (NO SEEK)
B2
EXCP VALIDITY CHECK ROUTINE
INTER9
C1 XVCK01 TEST VALIDITY OF CONTROL BLOCKS
C2 EXCP ENTRANCE
YES
NO
STAGE-2 EXIT EFFECTOR
C4 IEA0EF00 SCHEDULE TRANSIENT ERROR ROUTINE
02 K3 EXIT FROM EXCP SUPVSR VIA GENERAL RETURN VECTOR
F1
TO INTER7C TO ENQUEUE REQ
INT17C
D3 INITIALIZE LOGICAL QUEUE
02 D3 TO TEST CHAN MODULE (XCPTCH4) TO PREPARE TO RETRY I/O OPERATION
E2
TESTS IF IOB EXCEPTION FLAG (IOBEX) IS SET
INTER8
E1 IS ERROR ROUTINE IN CONTROL
NO
E2 DOES PERMANENT ERR. EXIST
NO
E3 RESET DCB 'ERROR ROUTINE IN CONTROL' BITS
07 A2 TO CHAN END APPENDAGE (INT019B)
YES
YES
F1
INTER7C
F1 ENQUEUE REQ AT TOP OF QUEUE
FROM 02C5
03 G2
07 A2
TO ABNORMAL END APPENDAGE (VIA INT031B)
07 A2
NO
YES
INT057
(CHART 11D3) INT058
G2 INTERR3A ERROR ROUTINE INTERFACE
TO INDICATE PERMANENT ERROR.
G3 RELATED REQUEST
YES
G4 PERM ERROR SET BY PREVIOUS REQUEST
NO
I/O PURGE ROUTINE
SCHEDULED BY STAGE 2 EXIT EFFECTOR (IEA0EF00) OF SUPERVISOR
H3 IGE0025E PURGE RELATED REQUESTS
H4 INDICATE ERROR RTN IS IN CONTROL
SET DCB EXCEPTION FLAG (DCBFL) TO '01'
04 G3 TO CHAN SEARCH (INT027) TO TRY TO RESTART CHANNEL

• Chart 04. General Flow Within I/O Interruption Supervisor


IECINT
FROM I/O FIRST-LEVEL INTERRUPTION HANDLER (IEAQIO00) AFTER AN I/O INTERRUPTION
A1 ENTRY
TESTS CSW FOR CHAN CONT CHK OR INTERFACE CONT CHK
B1 CCC OR ICC
NO
YES
B2 CHAN LAST START-ED ON THIS DEVICE
YES
NO
B3 GET LAST RQE ADDRESS
B5 PCI
NO
YES
UBC LOOKUP
INT001
C3 FIND THE UCB FOR THIS DEVICE
C5 PCI APPENDAGE
CCH/SER INTERFACE
SERR04
D1 PREPARE LINKAGE TO CHAN CHECK HANDLER OR SER
D2 EXIT TO SER RTN OR THE CHAN CHECK HANDLER
(SEE NOTE 1)
INT010A
D3 OTHER STATUS
YES
NO
D4 DEVICE END
NO
YES
INT005
D5 CHANNEL END
YES
NO
K2
E3
TRAP CODE MODULE
INTATT
ATTN. ROUTINE
E1 HANDLE ATTENTION CONDITION
E2 ATTN. COND
YES
NO
E3 INTTRAP DO DEVICE-DEPENDENT FUNCTIONS
INT010
E4 MAKE DEVICE AVAILABLE, CLR BUSY FLAG
04 F3
INT019 (CHART 12)
INTSEN
SENSE RTN
F1 ISSUE SENSE COMMAND, TEST RESULTS, STORE SENSE BYTES IN THE IOB
F2 UNIT CHECK
YES
NO
F3 CHANNEL END APPENDAGE
F4 ERROR CONDITION
NO
YES
H1
XCPPST
POST RTN (IEAOP101)
F5 PLACE NORMAL COMPLETION CODE IN ECB & READY THE REQUESTOR'S TASK
G3
(SEE CHART 02, BLOCKS G3, H3, J2)
04 G3
TRY AGAIN
INT031
G1 ABNORM END APPEND OR ERROR RTN INTERFACE
G2 PROG, PROTECTION, OR CHAN CONT. CHECK.
YES
NO
E3
INT027,INT030
G3 GET SEARCH CODE POINTER
G3
G4 RESULT OF SIO
J4
RQE ACCEPTED OR BUSY
G5 START I/O PROCEDURE
H1
ERROR ROUTINE INTERFACE
INTERR
H1 USE STAGE-2 EXIT EFFECTOR (IEAOEF00) TO SCHEDULE ERROR ROUTINE
G3
CHAN SEARCH
H3 INTCS GET NEXT RQE
H4 GET NEXT LOG CHAN QUEUE
CHANGE LCH VECTOR
NO
H5 CHANNEL BUSY
YES
J3 DETERMINE IF REQ CAN BE STARTED
YES
J4 LAST RQE ON LCH
NO
J4
YES
K2
INT033C
XCPCH1--XCPCH7
TEST CHAN MODULE
K1 RETURN TO TASK SUPERVISOR
K2 ENABLE AND THEN DISABLE INTERRUPTIONS
K3 END OF SEARCH
YES
NO
K4 CU BUSY
NO
K5 TEST IF CHAN IS BUSY
RETURN IS VIA LOC 'DISMISS' IN TYPE-1 INTERRUPTION HANDLER (IEAQIO00) AND THE DISPATCHER (IEAODS)

NOTE 1: THE RETURN TO THE I/O SUPERVISOR FROM THE CHAN CHECK HANDLER IS AT ANY OF THE FOLLOWING LOCATIONS:

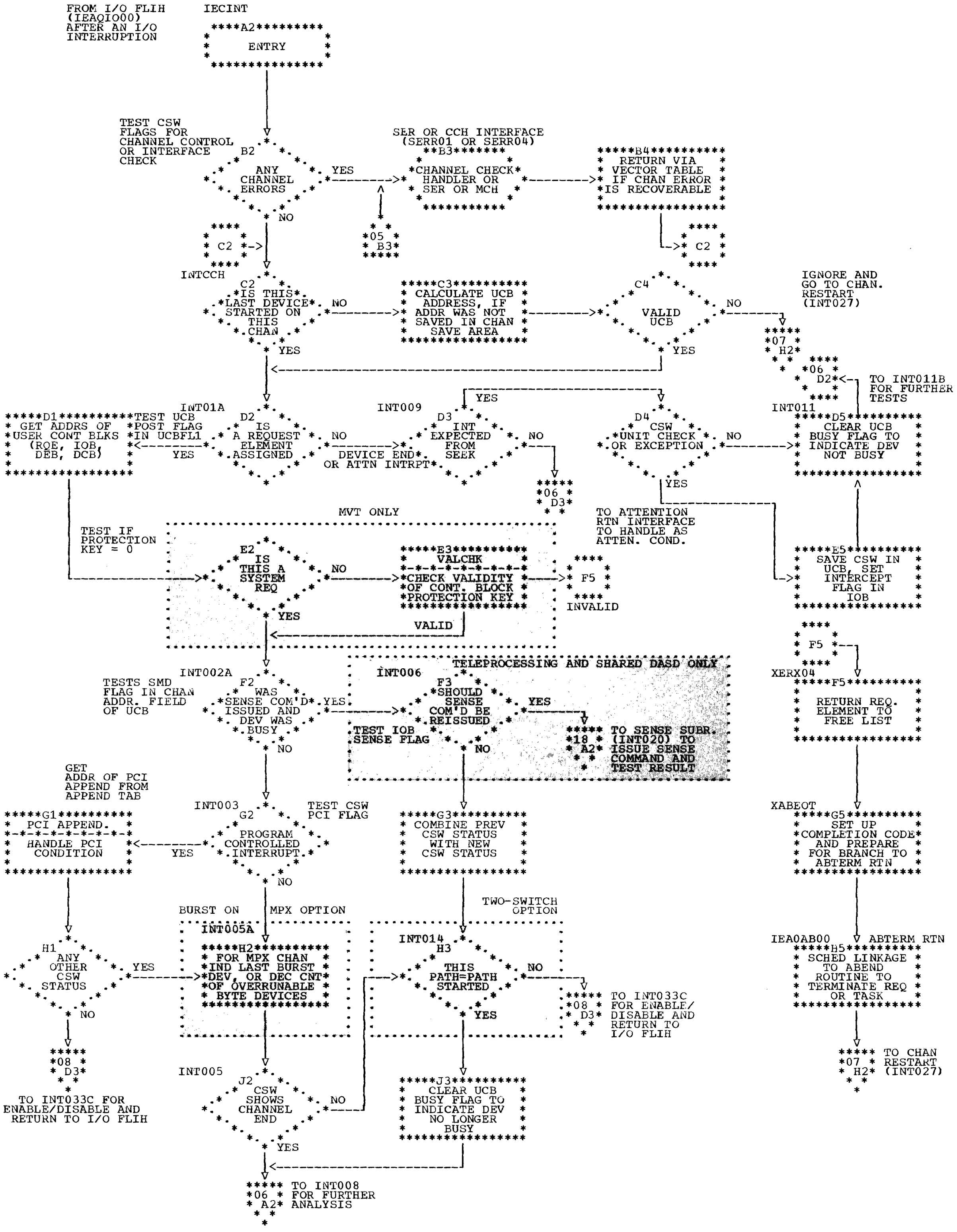
XCP056E IN POST SIO RTN - AFTER ISSUANCE OF AN SIO OR TIO INSTRUCTION FOR A SEEK OPERATION

INTCCH - AFTER AN INTERRUPTION

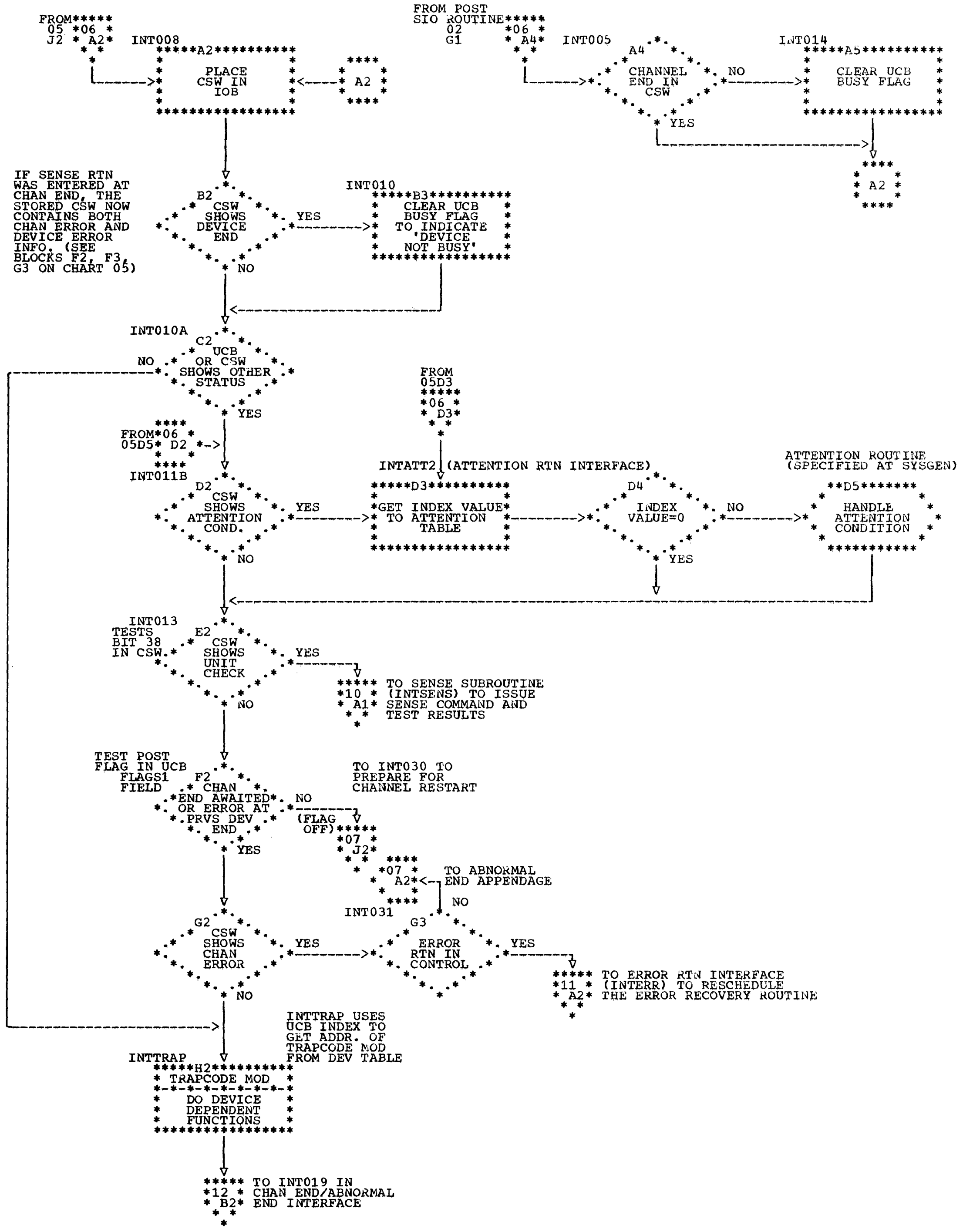
INTS03 IN THE SENSE SUBROUTINE - TO RETRY A SENSE COMMAND

INTS02A IN THE SENSE SUBROUTINE - AFTER CCH HAS ISSUED AN INVALID SENSE COMMAND. PERMITS A RETRY OF THE SENSE COMMAND.

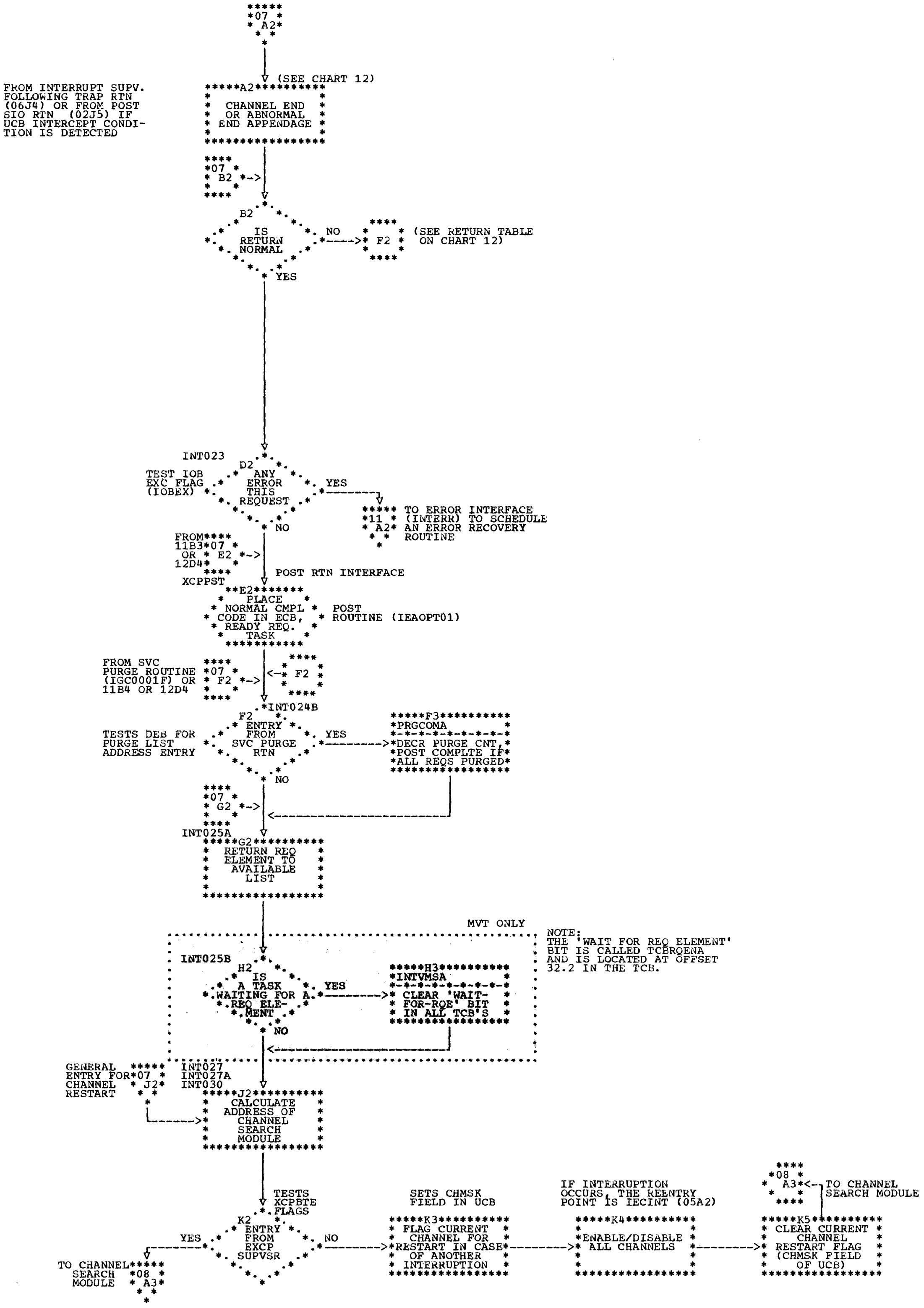
• Chart 05. I/O Interruption Supervisor (Part 1 of 8)


FROM I/O FLIH (IEAQIO00) AFTER AN I/O INTERRUPTION
IECINT
A2 ENTRY
TEST CSW FLAGS FOR CHANNEL CONTROL OR INTERFACE CHECK
B2 ANY CHANNEL ERRORS
YES
NO
SER OR CCH INTERFACE (SERR01 OR SERR04)
B3 CHANNEL CHECK HANDLER OR SER OR MCH
05 B3
B4 RETURN VIA VECTOR TABLE IF CHAN ERROR IS RECOVERABLE
C2
INTCCH
C2 IS THIS LAST DEVICE STARTED ON THIS CHAN
C3 CALCULATE UCB ADDRESS, IF ADDR WAS NOT SAVED IN CHAN SAVE AREA
C4 VALID UCB
IGNORE AND GO TO CHAN. RESTART (INT027)
07 H2
06 D2
TO INT011B FOR FURTHER TESTS
INT01A
D1 GET ADDRS OF USER CONT BLKS (RQE, IOB, DEB, DCB)
TEST UCB POST FLAG IN UCBFL1
D2 IS A REQUEST ELEMENT ASSIGNED
INT009
D3 INT EXPECTED FROM SEEK
DEVICE END OR ATTN INTRPT
06 D3
D4 CSW UNIT CHECK OR EXCEPTION
INT011
D5 CLEAR UCB BUSY FLAG TO INDICATE DEV NOT BUSY
TO ATTENTION RTN INTERFACE TO HANDLE AS ATTEN. COND.
MVT ONLY
TEST IF PROTECTION KEY = 0
E2 IS THIS A SYSTEM REQ
E3 VALCHK CHECK VALIDITY OF CONT. BLOCK PROTECTION KEY
F5
INVALID
VALID
E5 SAVE CSW IN UCB, SET INTERCEPT FLAG IN IOB
TELEPROCESSING AND SHARED DASD ONLY
INT002A
TESTS SMD FLAG IN CHAN ADDR. FIELD OF UCB
F2 WAS SENSE COM'D ISSUED AND DEV WAS BUSY
INT006
F3 SHOULD SENSE COM'D BE REISSUED
TEST IOB SENSE FLAG
18 A2 TO SENSE SUBR. (INT020) TO ISSUE SENSE COMMAND AND TEST RESULT
XERX04
F5 RETURN REQ. ELEMENT TO FREE LIST
GET ADDR OF PCI APPEND FROM APPEND TAB
G1 PCI APPEND. HANDLE PCI CONDITION
INT003
G2 PROGRAM CONTROLLED INTERRUPT
TEST CSW PCI FLAG
G3 COMBINE PREV CSW STATUS WITH NEW CSW STATUS
XABEOT
G5 SET UP COMPLETION CODE AND PREPARE FOR BRANCH TO ABTERM RTN
BURST ON
MPX OPTION
INT005A
H2 FOR MPX CHAN IND LAST BURST DEV, OR DEC CNT OF OVERRUNABLE BYTE DEVICES
TWO-SWITCH OPTION
INT014
H3 THIS PATH=PATH STARTED
H1 ANY OTHER CSW STATUS
08 D3 TO INT033C FOR ENABLE/ DISABLE AND RETURN TO I/O FLIH
IEAOAB00 ABTERM RTN
H5 SCHED LINKAGE TO ABEND ROUTINE TO TERMINATE REQ OR TASK
08 D3
TO INT033C FOR ENABLE/DISABLE AND RETURN TO I/O FLIH
INT005
J2 CSW SHOWS CHANNEL END
J3 CLEAR UCB BUSY FLAG TO INDICATE DEV NO LONGER BUSY
07 H2 TO CHAN RESTART (INT027)
06 A2 TO INT008 FOR FURTHER ANALYSIS

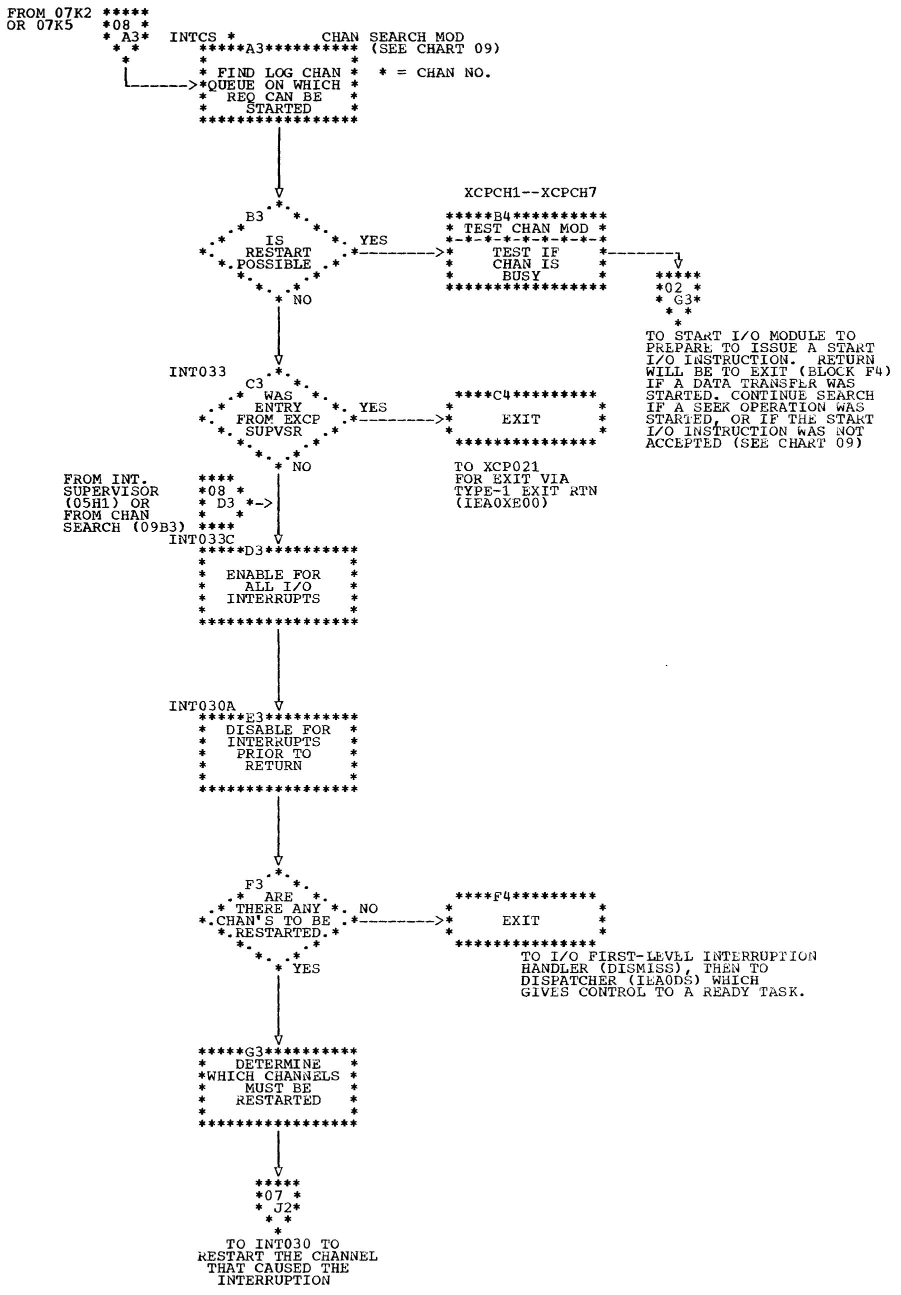
• Chart 06. I/O Interruption Supervisor (Part 2 of 8)


FROM 05 J2
06 A2
INT008
A2 PLACE CSW IN IOB
A2
FROM POST SIO ROUTINE 02 G1
06 A4
INT005
A4 CHANNEL END IN CSW
NO
YES
INT014
A5 CLEAR UCB BUSY FLAG
A2
IF SENSE RTN WAS ENTERED AT CHAN END, THE STORED CSW NOW CONTAINS BOTH CHAN ERROR AND DEVICE ERROR INFO. (SEE BLOCKS F2, F3, G3 ON CHART 05)
B2 CSW SHOWS DEVICE END
YES
NO
INT010
B3 CLEAR UCB BUSY FLAG TO INDICATE 'DEVICE NOT BUSY'
INT010A
C2 UCB OR CSW SHOWS OTHER STATUS
NO
YES
FROM 05D3
06 D3
FROM 05D5 06 D2
INT011B
D2 CSW SHOWS ATTENTION COND.
YES
NO
INTATT2 (ATTENTION RTN INTERFACE)
D3 GET INDEX VALUE TO ATTENTION TABLE
D4 INDEX VALUE=0
NO
YES
ATTENTION ROUTINE (SPECIFIED AT SYSGEN)
D5 HANDLE ATTENTION CONDITION
INT013 TESTS BIT 38 IN CSW.
E2 CSW SHOWS UNIT CHECK
YES
NO
10 A1 TO SENSE SUBROUTINE (INTSENS) TO ISSUE SENSE COMMAND AND TEST RESULTS
TEST POST FLAG IN UCB FLAGS1 FIELD
F2 CHAN END AWAITED OR ERROR AT PRVS DEV END
NO (FLAG OFF)
YES
TO INT030 TO PREPARE FOR CHANNEL RESTART
07 J2
07 A2 TO ABNORMAL END APPENDAGE
NO
INT031
G2 CSW SHOWS CHAN ERROR
YES
NO
G3 ERROR RTN IN CONTROL
YES
11 A2 TO ERROR RTN INTERFACE (INTERR) TO RESCHEDULE THE ERROR RECOVERY ROUTINE
INTTRAP USES UCB INDEX TO GET ADDR. OF TRAPCODE MOD FROM DEV TABLE
INTTRAP
H2 TRAPCODE MOD
DO DEVICE DEPENDENT FUNCTIONS
12 B2 TO INT019 IN CHAN END/ABNORMAL END INTERFACE

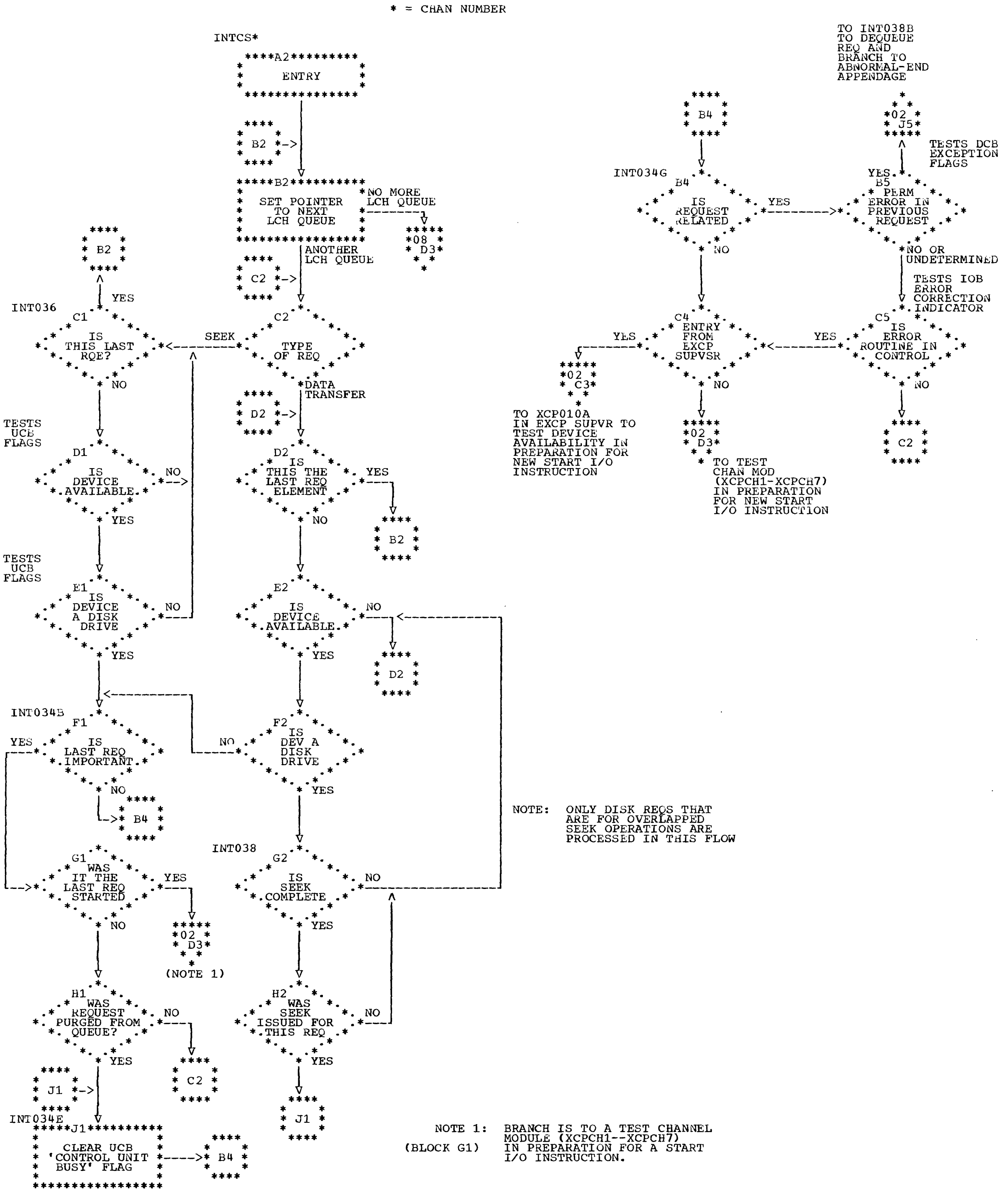
• Chart 07. I/O Interruption Supervisor (Part 3 of 8)


07 A2
(SEE CHART 12)
FROM INTERRUPT SUPV. FOLLOWING TRAP RTN (06J4) OR FROM POST SIO RTN (02J5) IF UCB INTERCEPT CONDITION IS DETECTED
A2
CHANNEL END OR ABNORMAL END APPENDAGE
07 B2
B2
IS RETURN NORMAL
NO
F2
(SEE RETURN TABLE ON CHART 12)
YES
INT023
TEST IOB EXC FLAG (IOBEX)
D2
ANY ERROR THIS REQUEST
YES
11 A2
TO ERROR INTERFACE (INTERR) TO SCHEDULE AN ERROR RECOVERY ROUTINE
NO
FROM 11B3 OR 12D4
07 E2
XCPPST
POST RTN INTERFACE
E2
PLACE NORMAL CMPL CODE IN ECB, READY REQ. TASK
POST ROUTINE (IEAOPT01)
FROM SVC PURGE ROUTINE (IGC0001F) OR 11B4 OR 12D4
07 F2
F2
INT024B
F2
ENTRY FROM SVC PURGE RTN
TESTS DEB FOR PURGE LIST ADDRESS ENTRY
YES
F3
PRGCOMA
DECR PURGE CNT, POST COMPLTE IF ALL REQS PURGED
NO
07 G2
INT025A
G2
RETURN REQ ELEMENT TO AVAILABLE LIST
MVT ONLY
NOTE: THE 'WAIT FOR REQ ELEMENT' BIT IS CALLED TCBRQENA AND IS LOCATED AT OFFSET 32.2 IN THE TCB.
INT025B
H2
IS A TASK WAITING FOR A REQ ELEMENT
YES
H3
INTVMSA
CLEAR 'WAIT-FOR-RQE' BIT IN ALL TCB'S
NO
GENERAL ENTRY FOR CHANNEL RESTART
07 J2
INT027
INT027A
INT030
J2
CALCULATE ADDRESS OF CHANNEL SEARCH MODULE
TESTS XCPBTE FLAGS
K2
ENTRY FROM EXCP SUPVSR
YES
TO CHANNEL SEARCH MODULE
08 A3
NO
SETS CHMSK FIELD IN UCB
K3
FLAG CURRENT CHANNEL FOR RESTART IN CASE OF ANOTHER INTERRUPTION
IF INTERRUPTION OCCURS, THE REENTRY POINT IS IECINT (05A2)
K4
ENABLE/DISABLE ALL CHANNELS
08 A3
TO CHANNEL SEARCH MODULE
K5
CLEAR CURRENT CHANNEL RESTART FLAG (CHMSK FIELD OF UCB)

• Chart 08. I/O Interruption Supervisor (Part 4 of 8)


FROM 07K2 OR 07K5
08 A3
INTCS
A3
CHAN SEARCH MOD (SEE CHART 09)
FIND LOG CHAN * QUEUE ON WHICH REQ CAN BE STARTED
* = CHAN NO.
B3
IS RESTART POSSIBLE
YES
NO
XCPCH1--XCPCH7
B4
TEST CHAN MOD
TEST IF CHAN IS BUSY
02 G3
TO START I/O MODULE TO PREPARE TO ISSUE A START I/O INSTRUCTION. RETURN WILL BE TO EXIT (BLOCK F4) IF A DATA TRANSFER WAS STARTED. CONTINUE SEARCH IF A SEEK OPERATION WAS STARTED, OR IF THE START I/O INSTRUCTION WAS NOT ACCEPTED (SEE CHART 09)
INT033
C3
WAS ENTRY FROM EXCP SUPVSR
YES
NO
C4
EXIT
TO XCP021 FOR EXIT VIA TYPE-1 EXIT RTN (IEA0XE00)
FROM INT. SUPERVISOR (05H1) OR FROM CHAN SEARCH (09B3)
08 D3
INT033C
D3
ENABLE FOR ALL I/O INTERRUPTS
INT030A
E3
DISABLE FOR INTERRUPTS PRIOR TO RETURN
F3
ARE THERE ANY CHAN'S TO BE RESTARTED
NO
YES
F4
EXIT
TO I/O FIRST-LEVEL INTERRUPTION HANDLER (DISMISS), THEN TO DISPATCHER (IEA0DS) WHICH GIVES CONTROL TO A READY TASK.
G3
DETERMINE WHICH CHANNELS MUST BE RESTARTED
07 J2
TO INT030 TO RESTART THE CHANNEL THAT CAUSED THE INTERRUPTION

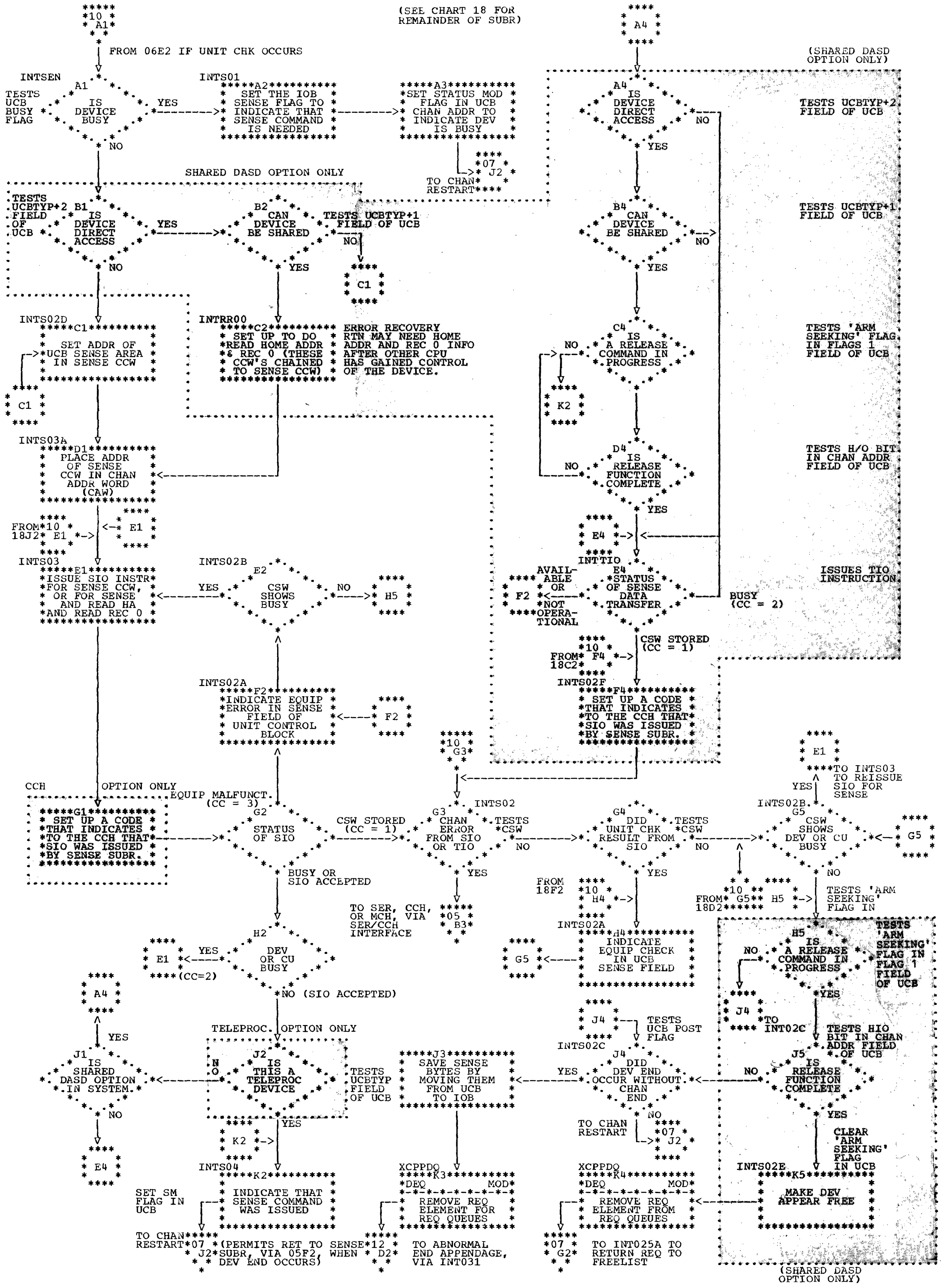
• Chart 09. I/O Interruption Supervisor (Part 5 of 8): Channel Search Module


* = CHAN NUMBER
INTCS*
A2
ENTRY
B2
B2
SET POINTER TO NEXT LCH QUEUE
NO MORE LCH QUEUE
08 D3
ANOTHER LCH QUEUE
C2
B2
YES
INT036
C1
IS THIS LAST RQE?
NO
SEEK
C2
TYPE OF REQ
DATA TRANSFER
D2
TESTS UCB FLAGS
D1
IS DEVICE AVAILABLE.
NO
YES
D2
IS THIS THE LAST REQ ELEMENT
YES
NO
B2
TESTS UCB FLAGS
E1
IS DEVICE A DISK DRIVE
NO
YES
E2
IS DEVICE AVAILABLE.
NO
YES
D2
INT034B
F1
IS LAST REQ IMPORTANT.
YES
NO
B4
F2
IS DEV A DISK DRIVE
NO
YES
INT038
G1
WAS IT THE LAST REQ STARTED
YES
NO
02 D3
(NOTE 1)
G2
IS SEEK COMPLETE
NO
YES
H1
WAS REQUEST PURGED FROM QUEUE?
NO
YES
C2
H2
WAS SEEK ISSUED FOR THIS REQ
NO
YES
J1
INT034E
J1
CLEAR UCB 'CONTROL UNIT BUSY' FLAG
B4
J1
TO INT038B TO DEQUEUE REQ AND BRANCH TO ABNORMAL-END APPENDAGE
B4
02 J5
TESTS DCB EXCEPTION FLAGS
INT034G
B4
IS REQUEST RELATED
YES
B5
PERM ERROR IN PREVIOUS REQUEST
YES
NO
NO OR UNDETERMINED
TESTS IOB ERROR CORRECTION INDICATOR
C4
ENTRY FROM EXCP SUPVSR
YES
C5
IS ERROR ROUTINE IN CONTROL
YES
02 C3
TO XCP010A IN EXCP SUPVR TO TEST DEVICE AVAILABILITY IN PREPARATION FOR NEW START I/O INSTRUCTION
NO
02 D3
TO TEST CHAN MOD (XCPCH1-XCPCH7) IN PREPARATION FOR NEW START I/O INSTRUCTION
NO
C2

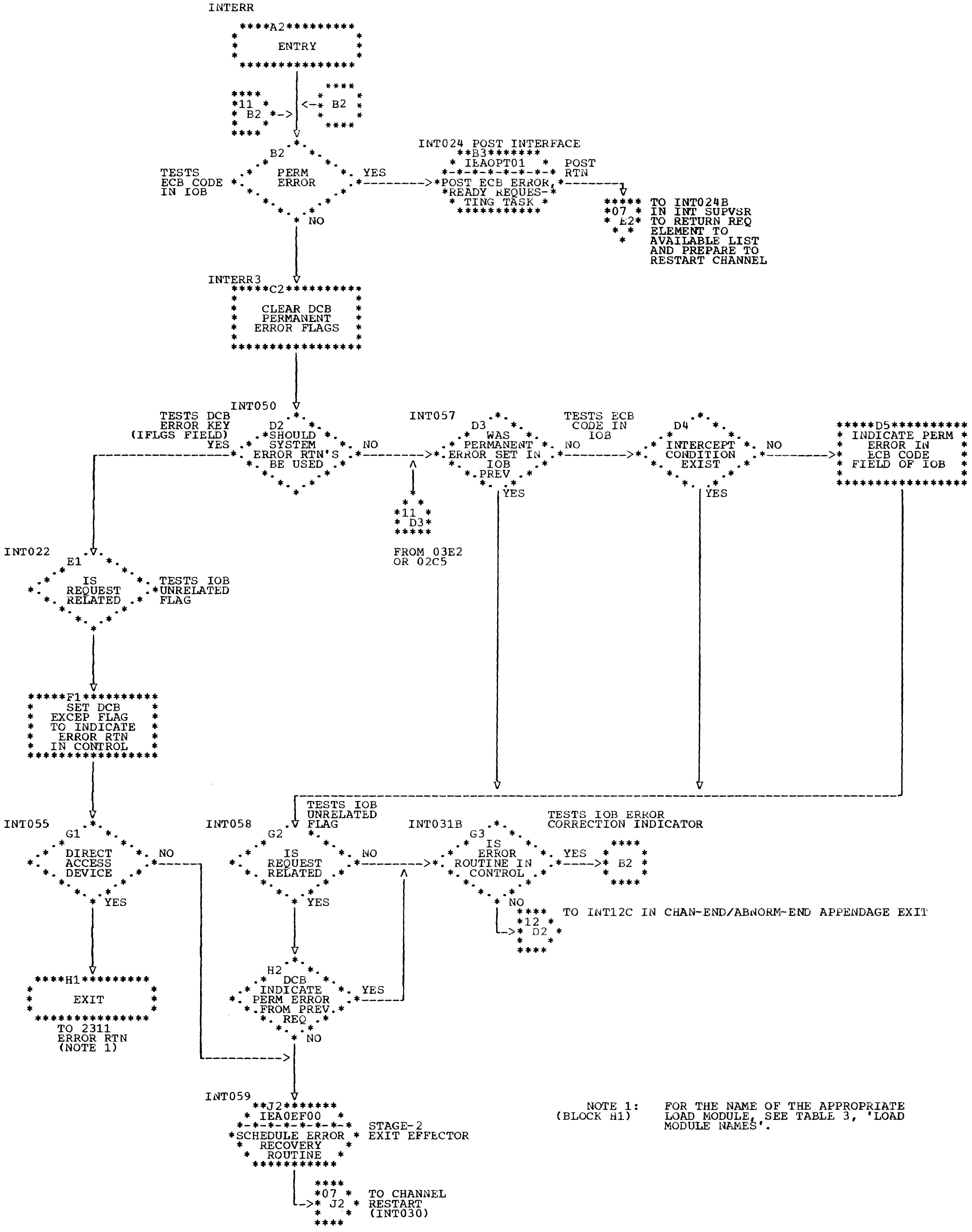
NOTE: ONLY DISK REQS THAT ARE FOR OVERLAPPED SEEK OPERATIONS ARE PROCESSED IN THIS FLOW

NOTE 1: BRANCH IS TO A TEST CHANNEL MODULE (XCPCH1--XCPCH7)
(BLOCK G1) IN PREPARATION FOR A START I/O INSTRUCTION.

• **Chart 10. I/O Interruption Supervisor (Part 6 of 8): Sense Subroutine (Part 1)**

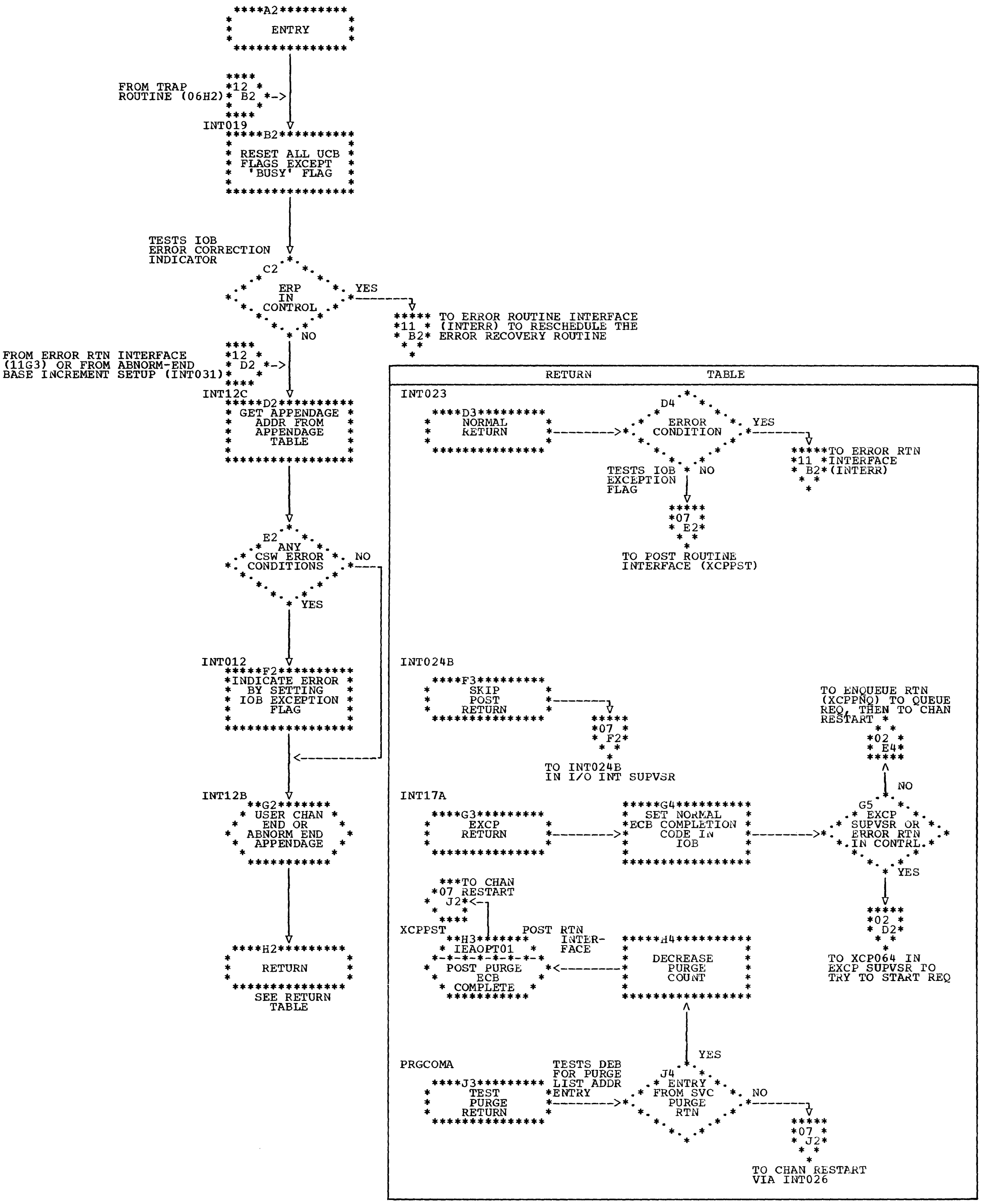

10 A1
FROM 06E2 IF UNIT CHK OCCURS
(SEE CHART 18 FOR REMAINDER OF SUBR)
A4
(SHARED DASD OPTION ONLY)
INTSEN A1
TESTS UCB BUSY FLAG
IS DEVICE BUSY
YES
NO
INTS01 A2
SET THE IOB SENSE FLAG TO INDICATE THAT SENSE COMMAND IS NEEDED
A3
SET STATUS MOD FLAG IN UCB CHAN ADDR TO INDICATE DEV IS BUSY
07 J2 TO CHAN RESTART
SHARED DASD OPTION ONLY
A4
IS DEVICE DIRECT ACCESS
NO
YES
TESTS UCBTYP+2 FIELD OF UCB
TESTS UCBTYP+2 FIELD OF UCB
B1
IS DEVICE DIRECT ACCESS
YES
NO
B2
CAN DEVICE BE SHARED
TESTS UCBTYP+1 FIELD OF UCB
NO
YES
C1
B4
CAN DEVICE BE SHARED
NO
YES
TESTS UCBTYP+1 FIELD OF UCB
INTS02D C1
SET ADDR OF UCB SENSE AREA IN SENSE CCW
C1
INTRR00 C2
SET UP TO DO READ HOME ADDR & REC 0 (THESE CCW'S CHAINED TO SENSE CCW)
ERROR RECOVERY RTN MAY NEED HOME ADDR AND REC 0 INFO AFTER OTHER CPU HAS GAINED CONTROL OF THE DEVICE.
C4
IS A RELEASE COMMAND IN PROGRESS
NO
K2
TESTS 'ARM SEEKING' FLAG IN FLAGS 1 FIELD OF UCB
INTS03A D1
PLACE ADDR OF SENSE CCW IN CHAN ADDR WORD (CAW)
D4
IS RELEASE FUNCTION COMPLETE
NO
YES
TESTS H/O BIT IN CHAN ADDR FIELD OF UCB
FROM 10 18J2 E1
E1
E4
INTS03 E1
ISSUE SIO INSTR FOR SENSE CCW, OR FOR SENSE AND READ HA AND READ REC 0
INTS02B E2
CSW SHOWS BUSY
YES
NO
H5
INTTIO E4
STATUS OF SENSE DATA TRANSFER
AVAILABLE OR NOT OPERATIONAL
F2
BUSY (CC = 2)
ISSUES TIO INSTRUCTION
CSW STORED (CC = 1)
FROM 10 18C2 F4
INTS02A F2
INDICATE EQUIP ERROR IN SENSE FIELD OF UNIT CONTROL BLOCK
F2
INTS02F F4
SET UP A CODE THAT INDICATES TO THE CCH THAT SIO WAS ISSUED BY SENSE SUBR.
10 G3
E1
TO INTS03 TO REISSUE SIO FOR SENSE
YES
CCH OPTION ONLY
EQUIP MALFUNCT. (CC = 3)
G1
SET UP A CODE THAT INDICATES TO THE CCH THAT SIO WAS ISSUED BY SENSE SUBR.
G2
STATUS OF SIO
CSW STORED (CC = 1)
BUSY OR SIO ACCEPTED
INTS02 G3
CHAN ERROR FROM SIO OR TIO
TESTS CSW
NO
YES
G4
DID UNIT CHK RESULT FROM SIO
TESTS CSW
NO
YES
INTS02B G5
CSW SHOWS DEV OR CU BUSY
G5
NO
FROM 10 18D2 G5 H5
TESTS 'ARM SEEKING' FLAG IN
TO SER, CCH, OR MCH, VIA SER/CCH INTERFACE
05 B3
FROM 18F2 10 H4
INTS02A H4
INDICATE EQUIP CHECK IN UCB SENSE FIELD
G5
H2
DEV OR CU BUSY
YES
E1
(CC=2)
NO (SIO ACCEPTED)
H5
IS A RELEASE COMMAND IN PROGRESS
NO
YES
TESTS 'ARM SEEKING' FLAG IN FLAG 1 FIELD OF UCB
A4
YES
J1
IS SHARED DASD OPTION IN SYSTEM.
NO
E4
TELEPROC. OPTION ONLY
J2
IS THIS A TELEPROC DEVICE
N O
YES
TESTS UCBTYP FIELD OF UCB
J3
SAVE SENSE BYTES BY MOVING THEM FROM UCB TO IOB
J4
TESTS UCB POST FLAG
INTS02C J4
DID DEV END OCCUR WITHOUT CHAN END
YES
NO
TO CHAN RESTART
07 J2
J4
TO INT02C
TESTS HIO BIT IN CHAN ADDR FIELD OF UCB
J5
IS RELEASE FUNCTION COMPLETE
NO
YES
CLEAR 'ARM SEEKING' FLAG IN UCB
K2
INTS04 K2
INDICATE THAT SENSE COMMAND WAS ISSUED
SET SM FLAG IN UCB
TO CHAN RESTART 07 J2
(PERMITS RET TO SENSE SUBR, VIA 05F2, WHEN DEV END OCCURS)
12 D2
XCPPDQ K3
DEQ MOD
REMOVE REQ ELEMENT FOR REQ QUEUES
TO ABNORMAL END APPENDAGE, VIA INT031
XCPPDQ K4
DEQ MOD
REMOVE REQ ELEMENT FROM REQ QUEUES
07 G2
TO INT025A TO RETURN REQ TO FREELIST
INTS02E K5
MAKE DEV APPEAR FREE
(SHARED DASD OPTION ONLY)

• Chart 11. I/O Interruption Supervisor (Part 7 of 8): Error Routine Interface

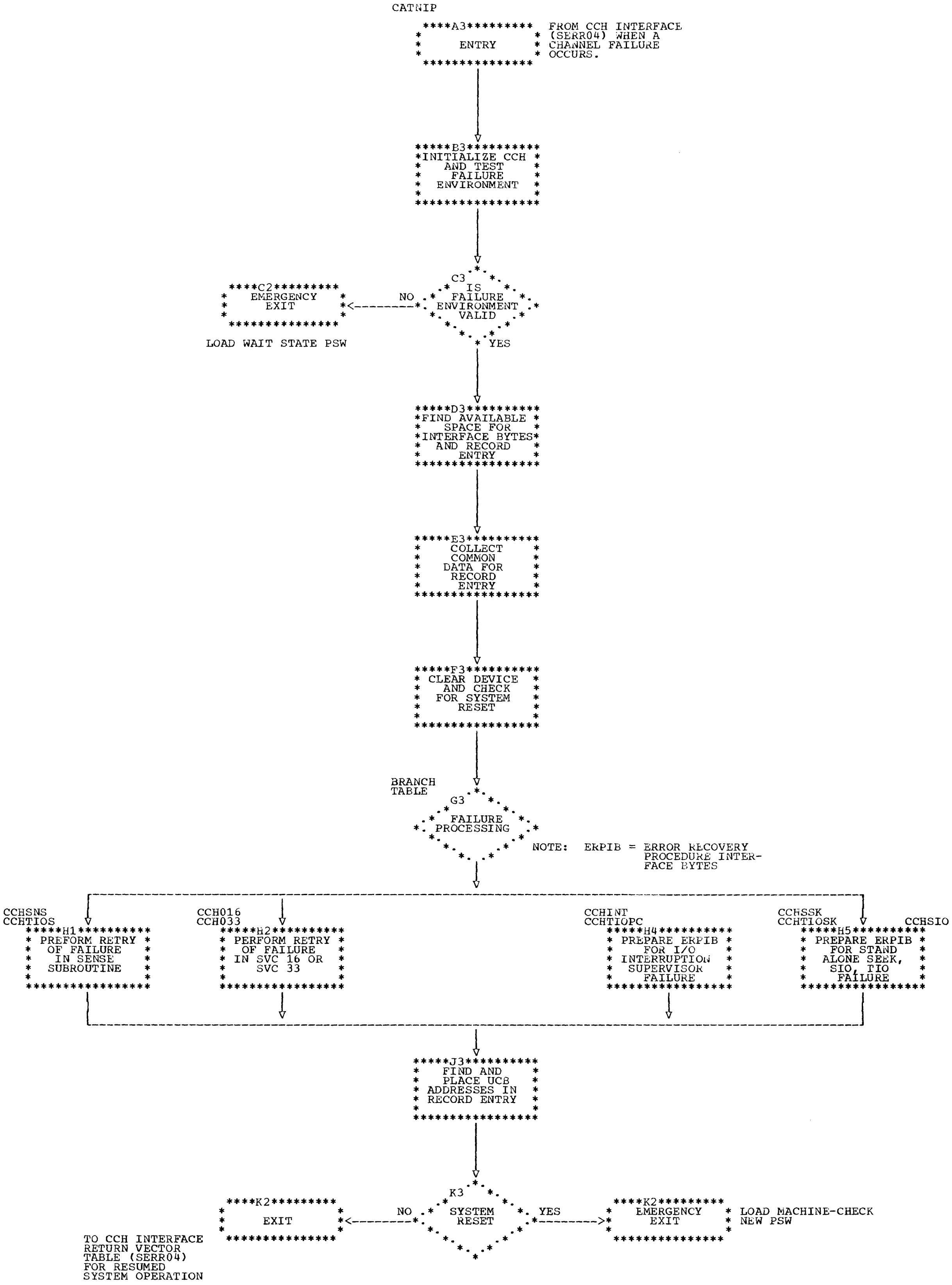

INTERR
A2
ENTRY
11 B2
B2
B2
TESTS ECB CODE IN IOB
PERM ERROR
YES
NO
INT024 POST INTERFACE
B3
IEAOPT01
POST ECB ERROR, READY REQUES-TING TASK
POST RTN
07 E2
TO INT024B IN INT SUPVSR TO RETURN REQ ELEMENT TO AVAILABLE LIST AND PREPARE TO RESTART CHANNEL
INTERR3
C2
CLEAR DCB PERMANENT ERROR FLAGS
INT050
TESTS DCB ERROR KEY (IFLGS FIELD)
D2
SHOULD SYSTEM ERROR RTN'S BE USED
YES
NO
INT057
D3
WAS PERMANENT ERROR SET IN IOB PREV
11 D3
FROM 03E2 OR 02C5
TESTS ECB CODE IN IOB
NO
YES
D4
INTERCEPT CONDITION EXIST
NO
YES
D5
INDICATE PERM ERROR IN ECB CODE FIELD OF IOB
INT022
E1
IS REQUEST RELATED
TESTS IOB UNRELATED FLAG
F1
SET DCB EXCEP FLAG TO INDICATE ERROR RTN IN CONTROL
INT055
G1
DIRECT ACCESS DEVICE
NO
YES
H1
EXIT
TO 2311 ERROR RTN (NOTE 1)
INT058
TESTS IOB UNRELATED FLAG
G2
IS REQUEST RELATED
NO
YES
INT031B
TESTS IOB ERROR CORRECTION INDICATOR
G3
IS ERROR ROUTINE IN CONTROL
YES
B2
NO
12 D2
TO INT12C IN CHAN-END/ABNORM-END APPENDAGE EXIT
H2
DCB INDICATE PERM ERROR FROM PREV. REQ
YES
NO
INT059
J2
IEA0EF00
SCHEDULE ERROR RECOVERY ROUTINE
STAGE-2 EXIT EFFECTOR
07 J2
TO CHANNEL RESTART (INT030)

NOTE 1: (BLOCK H1) FOR THE NAME OF THE APPROPRIATE LOAD MODULE, SEE TABLE 3, 'LOAD MODULE NAMES'.

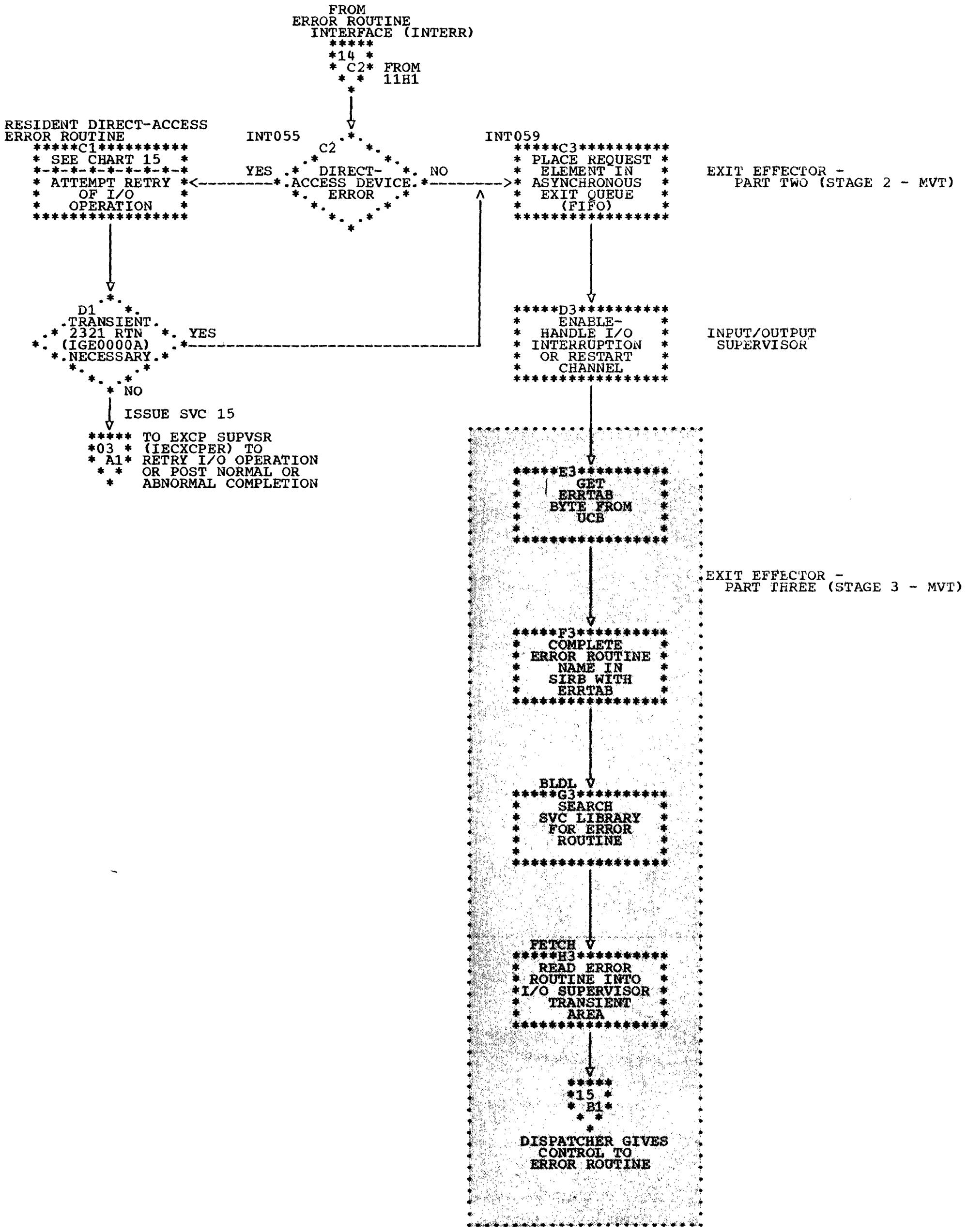
• Chart 12. I/O Interruption Supervisor (Part 8 of 8): Channel-End/Abnormal-End Appendage Interface


A2
ENTRY
FROM TRAP ROUTINE (06H2)
12 B2
INT019
B2
RESET ALL UCB FLAGS EXCEPT 'BUSY' FLAG
TESTS IOB ERROR CORRECTION INDICATOR
C2
ERP IN CONTROL
YES
NO
11 B2
TO ERROR ROUTINE INTERFACE (INTERR) TO RESCHEDULE THE ERROR RECOVERY ROUTINE
FROM ERROR RTN INTERFACE (11G3) OR FROM ABNORM-END BASE INCREMENT SETUP (INT031)
12 D2
INT12C
D2
GET APPENDAGE ADDR FROM APPENDAGE TABLE
E2
ANY CSW ERROR CONDITIONS
NO
YES
INT012
F2
INDICATE ERROR BY SETTING IOB EXCEPTION FLAG
INT12B
G2
USER CHAN END OR ABNORM END APPENDAGE
H2
RETURN
SEE RETURN TABLE
RETURN TABLE
INT023
D3
NORMAL RETURN
D4
ERROR CONDITION
YES
11 B2
TO ERROR RTN INTERFACE (INTERR)
TESTS IOB EXCEPTION FLAG
NO
07 E2
TO POST ROUTINE INTERFACE (XCPPST)
INT024B
F3
SKIP POST RETURN
07 F2
TO INT024B IN I/O INT SUPVSR
TO ENQUEUE RTN (XCPPNQ) TO QUEUE REQ, THEN TO CHAN RESTART
02 E4
NO
INT17A
G3
EXCP RETURN
G4
SET NORMAL ECB COMPLETION CODE IN IOB
G5
EXCP SUPVSR OR ERROR RTN IN CONTRL.
YES
02 D2
TO XCP064 IN EXCP SUPVSR TO TRY TO START REQ
07 J2
TO CHAN RESTART
XCPPST
H3
IEAOPT01
POST PURGE ECB COMPLETE
POST RTN INTER- FACE
H4
DECREASE PURGE COUNT
YES
PRGCOMA
J3
TEST PURGE RETURN
TESTS DEB FOR PURGE LIST ADDR ENTRY
J4
ENTRY FROM SVC PURGE RTN
NO
07 J2
TO CHAN RESTART VIA INT026

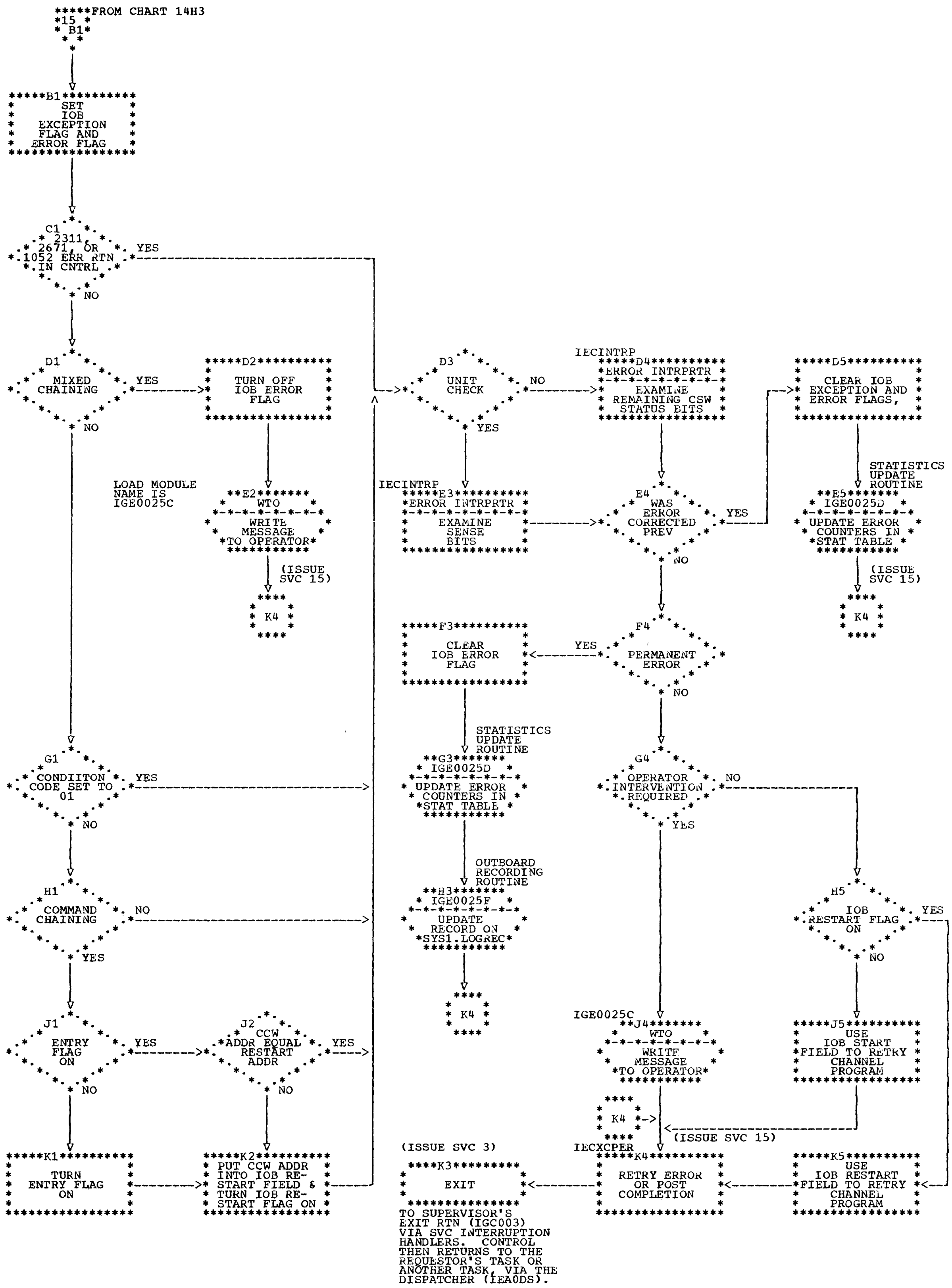
• Chart 13. Channel-Check Handler (CCH)


CATNIP
A3
ENTRY
FROM CCH INTERFACE (SERR04) WHEN A CHANNEL FAILURE OCCURS.
B3
INITIALIZE CCH AND TEST FAILURE ENVIRONMENT
C3
IS FAILURE ENVIRONMENT VALID
NO
C2
EMERGENCY EXIT
LOAD WAIT STATE PSW
YES
D3
FIND AVAILABLE SPACE FOR INTERFACE BYTES AND RECORD ENTRY
E3
COLLECT COMMON DATA FOR RECORD ENTRY
F3
CLEAR DEVICE AND CHECK FOR SYSTEM RESET
BRANCH TABLE
G3
FAILURE PROCESSING
NOTE: ERPIB = ERROR RECOVERY PROCEDURE INTERFACE BYTES
CCHSNS
CCHTIOS
H1
PREFORM RETRY OF FAILURE IN SENSE SUBROUTINE
CCH016
CCH033
H2
PERFORM RETRY OF FAILURE IN SVC 16 OR SVC 33
CCHINT
CCHTIOPC
H4
PREPARE ERPIB FOR I/O INTERRUPTION SUPERVISOR FAILURE
CCHSSK
CCHTIOSK
CCHSIO
H5
PREPARE ERPIB FOR STAND ALONE SEEK, SIO, TIO FAILURE
J3
FIND AND PLACE UCB ADDRESSES IN RECORD ENTRY
K3
SYSTEM RESET
NO
K2
EXIT
TO CCH INTERFACE RETURN VECTOR TABLE (SERR04) FOR RESUMED SYSTEM OPERATION
YES
K2
EMERGENCY EXIT
LOAD MACHINE-CHECK NEW PSW

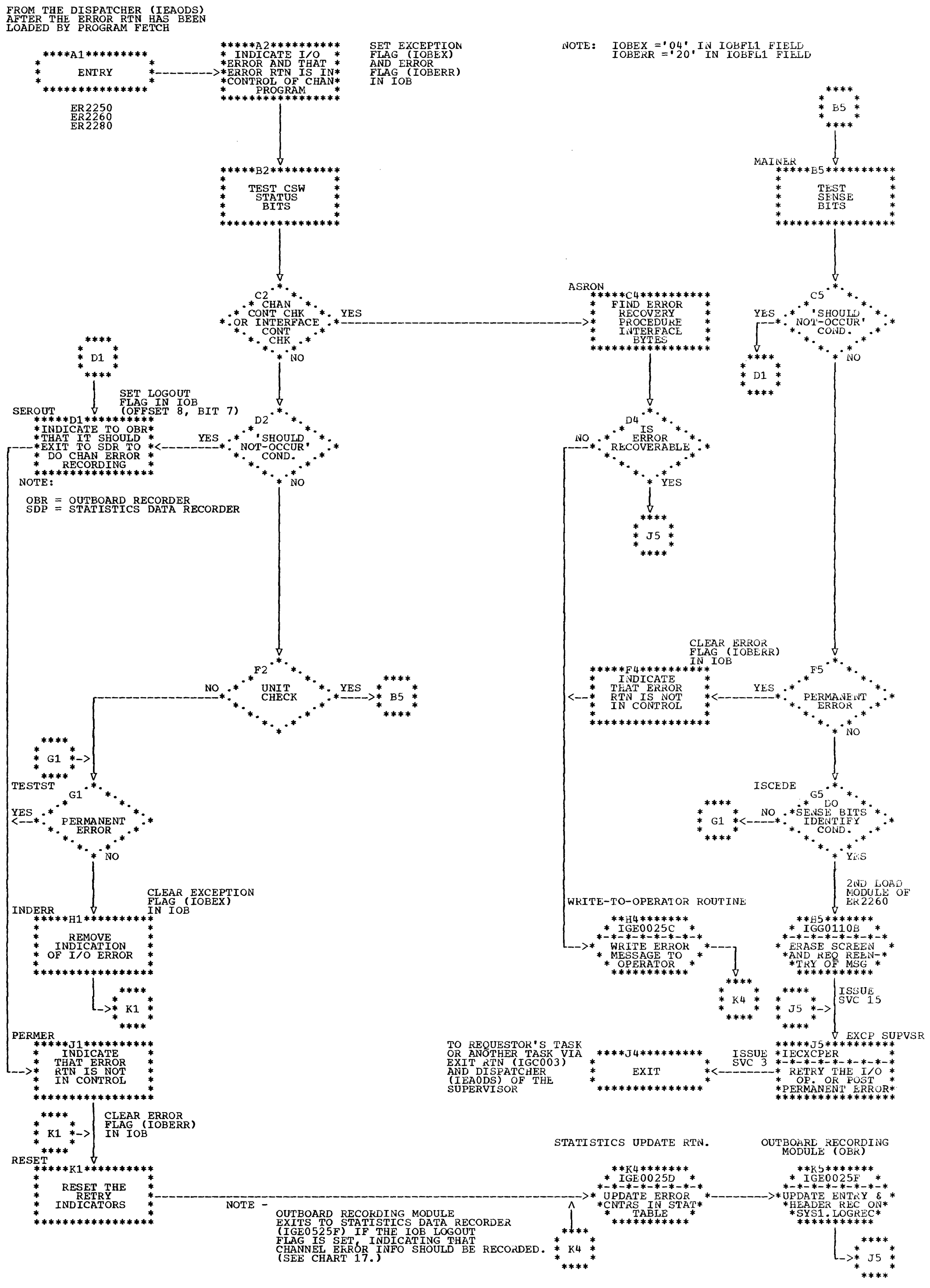
• Chart 14. Scheduling an IBM-Supplied Error Routine


FROM
ERROR ROUTINE
INTERFACE (INTERR)
14 C2
FROM 11H1
INT055
C2
DIRECT-ACCESS DEVICE ERROR
YES
NO
RESIDENT DIRECT-ACCESS ERROR ROUTINE
C1
SEE CHART 15
ATTEMPT RETRY OF I/O OPERATION
INT059
C3
PLACE REQUEST ELEMENT IN ASYNCHRONOUS EXIT QUEUE (FIFO)
EXIT EFFECTOR - PART TWO (STAGE 2 - MVT)
D1
TRANSIENT 2321 RTN (IGE0000A) NECESSARY
YES
NO
ISSUE SVC 15
03 A1
TO EXCP SUPVSR (IECXCPER) TO RETRY I/O OPERATION OR POST NORMAL OR ABNORMAL COMPLETION
D3
ENABLE-HANDLE I/O INTERRUPTION OR RESTART CHANNEL
INPUT/OUTPUT SUPERVISOR
E3
GET ERRTAB BYTE FROM UCB
EXIT EFFECTOR - PART THREE (STAGE 3 - MVT)
F3
COMPLETE ERROR ROUTINE NAME IN SIRB WITH ERRTAB
BLDL
G3
SEARCH SVC LIBRARY FOR ERROR ROUTINE
FETCH
H3
READ ERROR ROUTINE INTO I/O SUPERVISOR TRANSIENT AREA
15 B1
DISPATCHER GIVES CONTROL TO ERROR ROUTINE

• Chart 15. General Operating Procedure for Device-Dependent Error Routines

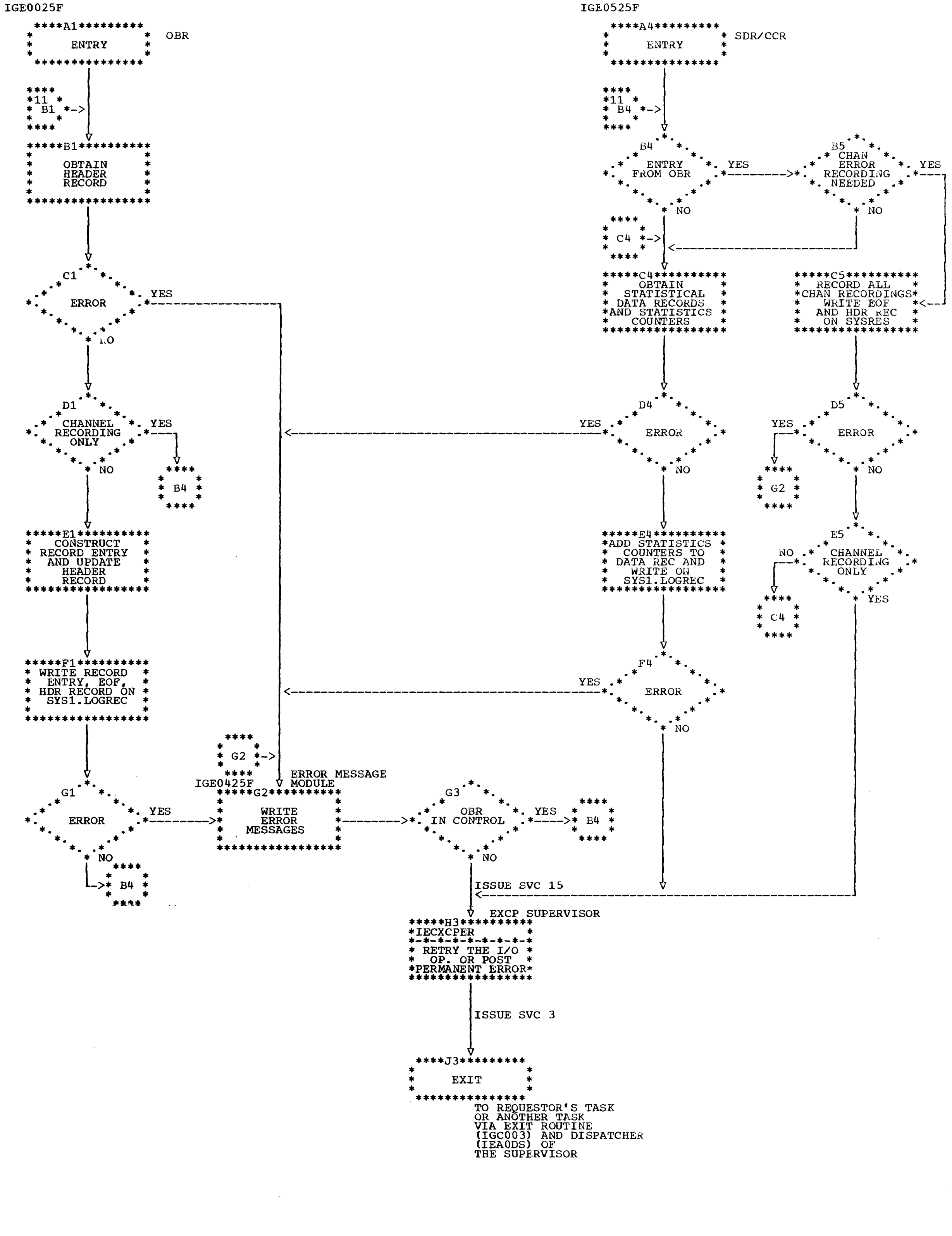

FROM CHART 14H3
15 B1
B1 SET IOB EXCEPTION FLAG AND ERROR FLAG
C1 2311, 2671, OR 1052 ERR RTN IN CNTRL
YES
NO
D1 MIXED CHAINING
YES
NO
D2 TURN OFF IOB ERROR FLAG
LOAD MODULE NAME IS IGE0025C
E2 WTO WRITE MESSAGE TO OPERATOR
(ISSUE SVC 15)
K4
D3 UNIT CHECK
NO
YES
IECINTRP
D4 ERROR INTRPRTR EXAMINE REMAINING CSW STATUS BITS
IECINTRP
E3 ERROR INTRPRTR EXAMINE SENSE BITS
E4 WAS ERROR CORRECTED PREV
YES
NO
D5 CLEAR IOB EXCEPTION AND ERROR FLAGS,
STATISTICS UPDATE ROUTINE
E5 IGE0025D UPDATE ERROR COUNTERS IN STAT TABLE
(ISSUE SVC 15)
K4
F3 CLEAR IOB ERROR FLAG
F4 PERMANENT ERROR
YES
NO
STATISTICS UPDATE ROUTINE
G3 IGE0025D UPDATE ERROR COUNTERS IN STAT TABLE
OUTBOARD RECORDING ROUTINE
H3 IGE0025F UPDATE RECORD ON SYS1.LOGREC
K4
G1 CONDIITON CODE SET TO 01
YES
NO
H1 COMMAND CHAINING
NO
YES
J1 ENTRY FLAG ON
YES
NO
J2 CCW ADDR EQUAL RESTART ADDR
YES
NO
K1 TURN ENTRY FLAG ON
K2 PUT CCW ADDR INTO IOB RESTART FIELD & TURN IOB RESTART FLAG ON
G4 OPERATOR INTERVENTION REQUIRED
NO
YES
IGE0025C
J4 WTO WRITE MESSAGE TO OPERATOR
K4
H5 IOB RESTART FLAG ON
YES
NO
J5 USE IOB START FIELD TO RETRY CHANNEL PROGRAM
(ISSUE SVC 15)
(ISSUE SVC 3)
K3 EXIT
IECXCPER
K4 RETRY ERROR OR POST COMPLETION
K5 USE IOB RESTART FIELD TO RETRY CHANNEL PROGRAM
TO SUPERVISOR'S EXIT RTN (IGC003) VIA SVC INTERRUPTION HANDLERS. CONTROL THEN RETURNS TO THE REQUESTOR'S TASK OR ANOTHER TASK, VIA THE DISPATCHER (IEA0DS).

• Chart 16. General Operating Procedure-Graphics


FROM THE DISPATCHER (IEAODS) AFTER THE ERROR RTN HAS BEEN LOADED BY PROGRAM FETCH
A1
ENTRY
ER2250
ER2260
ER2280
A2
INDICATE I/O ERROR AND THAT ERROR RTN IS IN CONTROL OF CHAN PROGRAM
SET EXCEPTION FLAG (IOBEX) AND ERROR FLAG (IOBERR) IN IOB
NOTE: IOBEX ='04' IN IOBFL1 FIELD
IOBERR ='20' IN IOBFL1 FIELD
B5
MAINER
B5
TEST SENSE BITS
B2
TEST CSW STATUS BITS
C2
CHAN CONT CHK OR INTERFACE CONT CHK
YES
NO
ASRON
C4
FIND ERROR RECOVERY PROCEDURE INTERFACE BYTES
C5
'SHOULD NOT-OCCUR' COND.
YES
NO
D1
D1
SET LOGOUT FLAG IN IOB (OFFSET 8, BIT 7)
SEROUT
D1
INDICATE TO OBR THAT IT SHOULD EXIT TO SDR TO DO CHAN ERROR RECORDING
NOTE:
OBR = OUTBOARD RECORDER
SDP = STATISTICS DATA RECORDER
D2
'SHOULD NOT-OCCUR' COND.
YES
NO
D4
IS ERROR RECOVERABLE
NO
YES
J5
CLEAR ERROR FLAG (IOBERR) IN IOB
F4
INDICATE THAT ERROR RTN IS NOT IN CONTROL
F5
PERMANENT ERROR
YES
NO
F2
UNIT CHECK
NO
YES
B5
G1
TESTST
G1
PERMANENT ERROR
YES
NO
ISCEDE
G5
DO SENSE BITS IDENTIFY COND.
NO
YES
G1
2ND LOAD MODULE OF ER2260
CLEAR EXCEPTION FLAG (IOBEX) IN IOB
INDERR
H1
REMOVE INDICATION OF I/O ERROR
WRITE-TO-OPERATOR ROUTINE
H4
IGE0025C
WRITE ERROR MESSAGE TO OPERATOR
H5
IGG0110B
ERASE SCREEN AND REQ REEN-TRY OF MSG
K1
K4
J5
ISSUE SVC 15
PERMER
J1
INDICATE THAT ERROR RTN IS NOT IN CONTROL
TO REQUESTOR'S TASK OR ANOTHER TASK VIA EXIT RTN (IGC003) AND DISPATCHER (IEAODS) OF THE SUPERVISOR
J4
EXIT
ISSUE SVC 3
EXCP SUPVSR
J5
IECXCPER
RETRY THE I/O OP. OR POST PERMANENT ERROR
K1
CLEAR ERROR FLAG (IOBERR) IN IOB
RESET
K1
RESET THE RETRY INDICATORS
STATISTICS UPDATE RTN.
K4
IGE0025D
UPDATE ERROR CNTRS IN STAT TABLE
OUTBOARD RECORDING MODULE (OBR)
K5
IGE0025F
UPDATE ENTRY & HEADER REC ON SYS1.LOGREC
NOTE - OUTBOARD RECORDING MODULE EXITS TO STATISTICS DATA RECORDER (IGE0525F) IF THE IOB LOGOUT FLAG IS SET, INDICATING THAT CHANNEL ERROR INFO SHOULD BE RECORDED. (SEE CHART 17.)
K4
J5

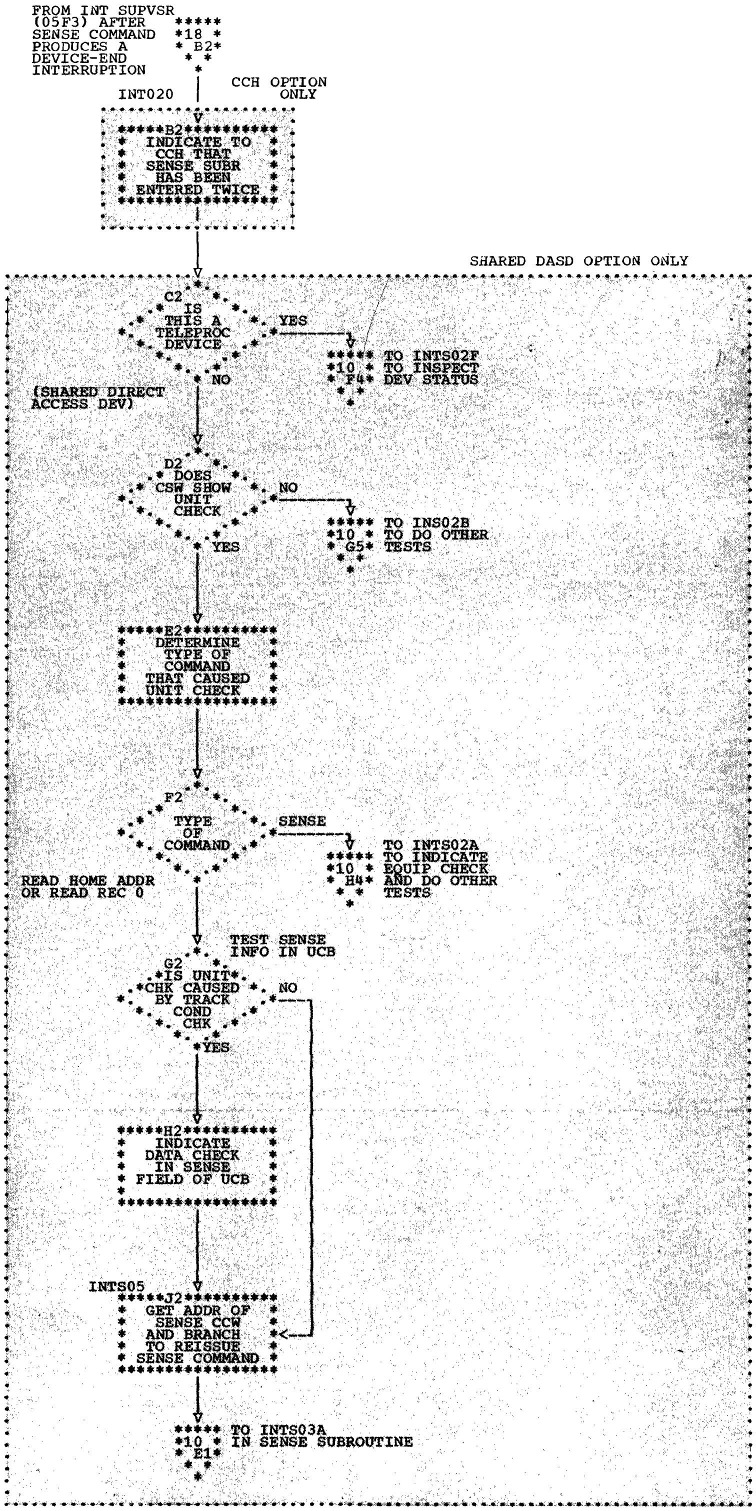
]

• Chart 17. Onboard Recorder and Statistical Data Recorder/Channel Check Recorder


IGE0025F
A1
ENTRY
OBR
11
B1
B1
OBTAIN HEADER RECORD
C1
ERROR
YES
NO
D1
CHANNEL RECORDING ONLY
YES
NO
B4
E1
CONSTRUCT RECORD ENTRY AND UPDATE HEADER RECORD
F1
WRITE RECORD ENTRY, EOF, HDR RECORD ON SYS1.LOGREC
G1
ERROR
YES
NO
B4
G2
IGE0425F
ERROR MESSAGE MODULE
G2
WRITE ERROR MESSAGES
G3
OBR IN CONTROL
YES
B4
NO
ISSUE SVC 15
EXCP SUPERVISOR
H3
IECXCPER
RETRY THE I/O OP. OR POST PERMANENT ERROR
ISSUE SVC 3
J3
EXIT
TO REQUESTOR'S TASK OR ANOTHER TASK VIA EXIT ROUTINE (IGC003) AND DISPATCHER (IEAODS) OF THE SUPERVISOR
IGE0525F
A4
ENTRY
SDR/CCR
11
B4
B4
ENTRY FROM OBR
YES
NO
B5
CHAN ERROR RECORDING NEEDED
YES
NO
C4
C4
OBTAIN STATISTICAL DATA RECORDS AND STATISTICS COUNTERS
C5
RECORD ALL CHAN RECORDINGS WRITE EOF AND HDR REC ON SYSRES
D4
ERROR
YES
NO
D5
ERROR
YES
NO
G2
E4
ADD STATISTICS COUNTERS TO DATA REC AND WRITE ON SYS1.LOGREC
E5
CHANNEL RECORDING ONLY
NO
YES
C4
F4
ERROR
YES
NO

• Chart 18. Sense Subroutine (Part 2)

(SEE CHART 10 FOR OTHER SECTION OF SENSE SUBROUTINE.)


FROM INT SUPVSR (05F3) AFTER SENSE COMMAND PRODUCES A DEVICE-END INTERRUPTION
18 B2
INT020
CCH OPTION ONLY
B2
INDICATE TO CCH THAT SENSE SUBR HAS BEEN ENTERED TWICE
SHARED DASD OPTION ONLY
C2
IS THIS A TELEPROC DEVICE
YES
10 F4
TO INTS02F TO INSPECT DEV STATUS
NO
(SHARED DIRECT ACCESS DEV)
D2
DOES CSW SHOW UNIT CHECK
NO
10 G5
TO INS02B TO DO OTHER TESTS
YES
E2
DETERMINE TYPE OF COMMAND THAT CAUSED UNIT CHECK
F2
TYPE OF COMMAND
SENSE
10 H4
TO INTS02A TO INDICATE EQUIP CHECK AND DO OTHER TESTS
READ HOME ADDR OR READ REC 0
TEST SENSE INFO IN UCB
G2
IS UNIT CHK CAUSED BY TRACK COND CHK
NO
YES
H2
INDICATE DATA CHECK IN SENSE FIELD OF UCB
INTS05
J2
GET ADDR OF SENSE CCW AND BRANCH TO REISSUE SENSE COMMAND
10 E1
TO INTS03A IN SENSE SUBROUTINE

Appendix A: Load Module Names

The load module names for the I/O supervisor, the error routines, and the purge and restore routines are shown in Table 3.

Table 3. Load Module Names (Part 1 of 2)

Load Module Name	Routine Name
IEAASU00	Resident I/O Supervisor and Fixed-Task Supervisor
IEC23XXB	Resident Error Routine for 2311 and 2321 (without overflow)
IEC23XXC	Resident Error Routine for 2311 and 2321 (with overflow)
IEC23XXD	Resident Error Routine for 2311 and/or 2301,2302,2303,2314,2321 (without overflow)
IEC23XXE	Resident Error Routine for 2311 and/or 2301,2302,2303,2314,2321 (with overflow)
IGE0000A	Transient Direct Access Device Error Routine
IGE0001C	Load 1 of 2540/2821 Error Routine
IGE0101C	Load 2
IGE0000D	1052,2150 Error Routine
IGE0000E	1442,2501,2520 Error Routine
IGE0000G	1403/1443 Error Routine
IGE0002I	2671/2822 Error Routine
IGE0002H	Load 1 of Transient 2321 Error Routine
IGE0102H	Load 2
IGE0202H	Load 3
IGE0302H	Load 4
IGE0402H	Load 5
IGE0000I	Load 1 of 2400 Tape Series Error Routine
IGE0100I	Load 2
IGE0200I	Load 3
IGE0900I	Load 4
IGE0010A	2250 Error Routine
IGE0010B	Load 1 of 2260 Error Routine
IGE0110B	Load 2
IGE0010C	2280/82 Error Routine
IECINTRP	Resident Error Interpreter Routine
IGE0025C	Write-to-Operator Routine
IGE0025D	Statistical Update Routine
IGE0025E	I/O Purge Routine
IGE0025F	OBR routine

(Part 1 of 2)

Table 3. Load Module Names (Part 2 of 2)

Load Module Name	Routine Name
IGE0425F	Error Message Module
IGE0525F	Load 1 of SDR Routine
IGC0001F	SVC Purge Routine
IGC0001G	SVC Restore Routine
IGC0002D	SVC DEVTYPE Routine
IGC0003C	SVC IOHALT Routine
IGC0007F	SYS1.LOGREC Data Set Statistics Update Routine

(Part 2 of 2)

Appendix B: Linkages Within the Request Element Table

The request element table contains request elements that represent active I/O requests, available request elements that form a freelist, and queued request elements that are part of a logical channel queue. The following example provides a detailed explanation of these elements and their linkages within the request element table.

In Figure 23, the request element table contains fourteen elements: four available request elements are in the freelist, two active request elements represent active I/O requests and eight (plus one active with a stand alone seek) queued request elements are in a logical channel queue.

- Available request elements. The freelist pointer contains location 2132, which is the address of the next available request element (first element in the freelist). The request element at location 2132 points to the element at location 2144, which is linked through location 2060, the address of the last available element in the freelist. The last available element is identified by the dummy address of hexadecimal FFFF in its link field.

- Active request elements. The unit control blocks that represent the two devices engaged in active I/O operations contain the addresses of the request elements that represent the I/O requests for the operations. Unit Control Block 1 identifies the request element at location 2108 as the representative of one active I/O request. Unit Control Block 2 identifies the request element at location 2036 as the representative of an active I/O request which has a seek in process on a direct access device. This request element is also chained in the logical channel queue for the duration of the stand alone seek.

- Logical channel queue. The logical channel word for the logical channel queue identifies the first and last request elements in the queue. The first request element in the queue resides at location 2024 and this element is linked through location 2084, the address of the last element in the queue. The last request element is identified by the dummy address FFFF contained in its link field.

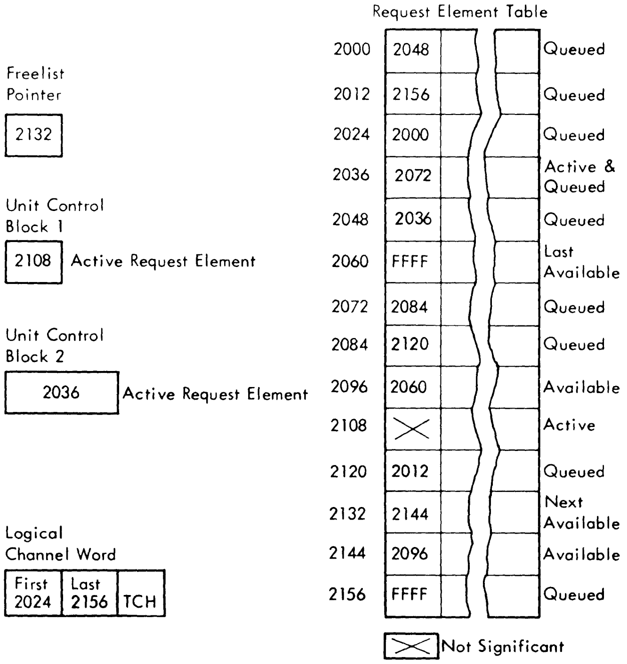


Figure 23. Linkages Within the Request Element Table

Appendix C: Theory of Logical Channels

Data that is transferred from the central processing unit to an I/O device follows a path through a physical channel. An I/O device can be switched among several physical channels, each of which provides a separate path to the device. The unique collection of all physical channel paths to a particular device is called a logical channel. If more than one device is included in the same set of physical channel paths, those devices are included in the same logical channel.

A physical channel is included in one or more logical channels, although a particular I/O device is included in only one logical channel. In Figure 24, for example, two logical channels (A and B) are shown. Logical channel A consists of the paths through physical channels 1 and 2 to device A, and logical channel B consists of the path through physical channel 2 to devices B and C. The devices and physical channel 1 are included in only one logical channel, whereas physical channel 2 is included in both logical channels.

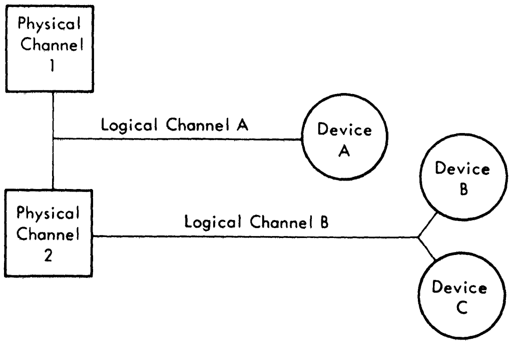


Figure 24. Logical Channels

The physical channel path to a particular I/O device may vary because of channel switching and subchannel considerations. Channel switching occurs when a device is switched, for operation purposes, from one physical channel to one of the other physical channels to which it is attached. In Figure 24, device A may be switched between physical channels 1 and 2. Channel switching does not hold for a multiplexor channel.

A subchannel is the physical channel facility required for a single data transfer operation between the CPU and an I/O device.

A selector channel has only one subchannel and the attached devices share the subchannel. A selector channel is used for only one data transfer operation at a time; however, other I/O devices attached to the selector channel can be simultaneously executing control operations. In Figure 24, if physical channel 2 was a selector channel, device B could maintain a data transfer operation while device C was executing a control operation, and vice versa.

A multiplexor channel, although having only one logical channel, has more than one subchannel, and the devices attached to it can sustain concurrently one I/O operation for each subchannel. In Figure 24, if physical channel 2 was a multiplexor channel, devices B and C could operate simultaneously. Note that device A could not operate because channel switching does not apply for a multiplexor channel, and because a multiplexor channel has only one logical channel.

Various physical channel paths and their logical channels are shown in Figure 25 and are described in the following list. Assume that physical channel 0 is a multiplexor channel and that physical channels 1 through 5 are selector channels.

- Logical Channel A is defined for the paths from physical channel 0 and its subchannels to the tape units on control unit 0 (the tape units share a subchannel and operate one at a time); to the reader, punch, and printer, respectively, on the integrated control unit (the reader, punch, and printer each use a separate subchannel and operate simultaneously); to two of the telecommunications lines on the transmission control unit (each telecommunications line uses a separate subchannel).

- Logical Channel B is defined for the path from physical channel 1 to the I/O devices on control unit 1.

- Logical Channel C is defined for the paths from physical channels 1 and 2 to the I/O devices on control units 2 and 3. Because control units 2 and 3 use the same physical channel paths, each device attached to these control units is in the same logical channel.

- Logical Channel D is defined for the paths from physical channels 1 and 3 to the I/O devices on control unit 4.

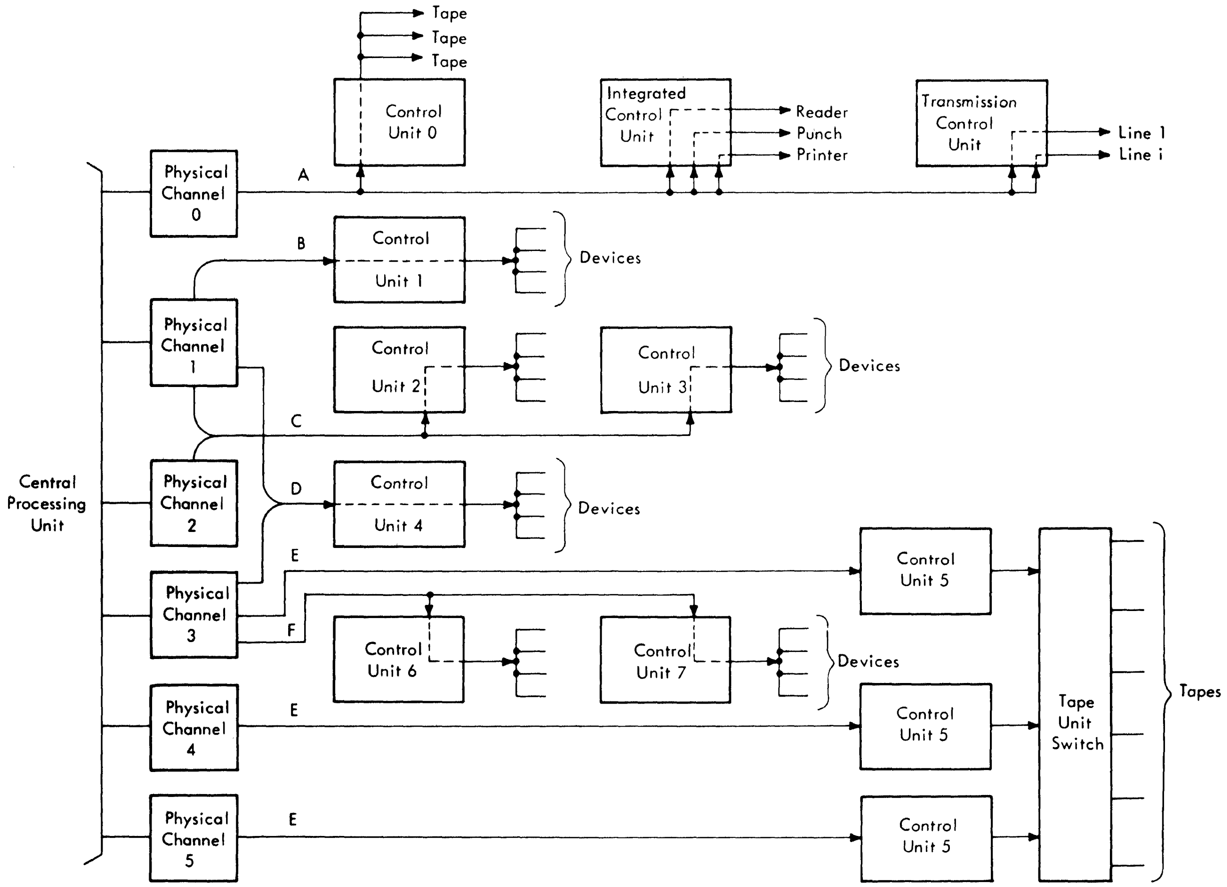


Figure 25. Physical Channel Paths and Logical Channels

- Logical Channel E is defined for the paths from physical channels 3, 4, and 5 to the tape units attached to the tape unit switch.

  Note: All control units on a tape unit switch must be in the same relative position on their respective physical channels (e.g., position 5 in Figure 25). This is the only unit address assignment requirement. The I/O supervisor requires such positioning for efficient interruption handling.

- Logical Channel F is defined for the path from physical channel 3 to the I/O devices on control units 6 and 7. Because the same physical channel paths are used for control units 6 and 7, each device attached to these control units is in the same logical channel.

Appendix D: System Environment Recording Output Data Set Format

The System Environment Recording (SER) routines require a data extent upon the system residence device; this data extent is the SER output data set (SYS1.LOGREC), which is formatted and initialized by the SYS1. LOGREC initialization program. For further information refer to the Utilities Program Logic Manual.

The SER output data set is composed, in contiguous positions, of the following types of records:

- Header record
- Statistical data records
- Record entries

HEADER RECORD

The header record is the first record on the first track of the extent and is preceded by a 2-byte key of hexadecimal FFFF. The header record describes the entire extent of the SER output data set.

The header record has the following characteristics:

- Creation. The header record is created at system generation time by the SYS1.LOGREC initialization program.
- Storage Area. It resides on the system residence device.
- Size. The header record is 38 bytes in length.
- Format. The format of the header record is shown in Figure 26.

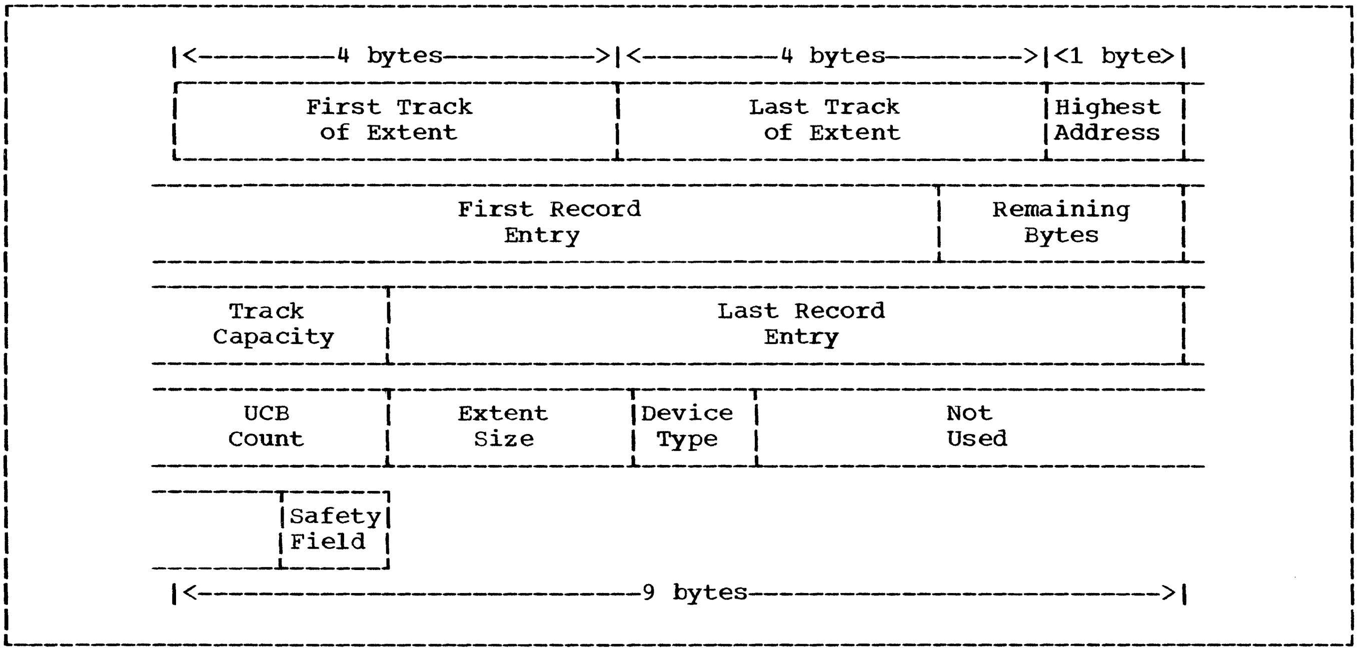


Figure 26. Header Record Format

Extent Limits (8 bytes)
contains the addresses (CCHH) of the first and last tracks of the data set.

Highest Address (1 byte)
contains the highest address of a track on a cylinder for the particular system residence device.

First Record Entry (7 bytes)
contains the address (BBCCHHR) and ID of the first track of the record entry area. Initially, the ID is zero.

Remaining Bytes (2 bytes)
contains the number of bytes remaining on the track upon which the last record entry was written. Initially, this field contains the track capacity.

Track Capacity (2 bytes)
contains the number of bytes per track for the particular system residence device.

Last Record Entry (7 bytes)
contains the address (BBCCHHR) and ID of the track of the last record entry written in the extent.

UCB Count (2 bytes)
contains the number of UCBs in the system.

Extent Size (2 bytes)
contains the number of tracks in the data set extent.

Device Type (1 byte)
contains an encoded field that designates the type of system residence device as follows:

Device	Code
2311	1
2301	2
2314	8

Safety Field (1 byte)
should contain hexadecimal FF. If an overrun condition, caused by a machine check or channel inboard failure, occurred while the header record was being written, the header record is padded with zeros from the point of overrun. Thus, if this field is zero, critical data has been destroyed and safety precautions are taken to halt and prevent further recording attempts.

STATISTICAL DATA RECORDS

The statistical data records follow the header record. Each is preceded by a 2-byte key. The keys are sequentially numbered, starting with zero, and are the index values to the statistics table entries associated with the particular statistical data record.

The statistical data records have the following characteristics:

- Creation. The extent is formatted by the SYS1.LOGREC initialization program and each record is created and written by SDR.
- Storage Area. The statistical data records reside on the system residence device.
- Quantity. There is one statistical data record for each device in the system.
- Size. Each statistical data record is 38 bytes in length and is preceded by a 2-byte key.
- Format. The format of a statistical data record is shown in Figure 27.

Channel and Device Address	Device Type	Counter		Counter
<--2 bytes--->	<----------4 bytes---------->	<-------32 bytes------->		

Figure 27. Statistical Data Record Format

Channel and Device Address (2 bytes)
contains the address of the unit associated with this particular statistical data record.

Device Type (4 bytes)
contains an encoded device type field for the device associated with the statistical data record. The field is the same as the UCB device type field.

Counters (32 bytes)
contains sixteen 2-byte counters. Each 4-bit counter within the statistics table of the I/O supervisor is expanded to a 2-byte counter on the statistical data record.

These sixteen counters are the summaries of the 4-bit counters and provide a device error history.

RECORD ENTRIES

The record entry area starts on the next consecutive track after the last statistical data record. The OBR writes into it I/O outboard fixed-length record entries containing information pertinent to the type of device in error, and I/O inboard record entries.

A record entry has the following characteristics:

- Creation. The area is allocated by the SYS1.LOGREC initialization program and each record is created and written by OBR.
- Storage Area. The record entries reside on the system residence device.
- Size. The length of the record entry area is fixed for each type of system residence device. It depends upon the number of tracks required to receive approximately 100 I/O outboard record entries.
- Format. The format of an I/O outboard record entry is shown in Figure 28. The format of an I/O inboard record entry produced by CCH is shown in Figure 29.

```
|<-----------4 bytes----------->|<1 byte>|<1 byte>|<--2 bytes--->|
 --------------------------------------------------------------
|  S       E       R     |   b   |Not Used|RE Type |  Not Used   |
 --------------------------------------------------------------
 --------------------------------------------------------------
               Date             |             Time             |
 --------------------------------------------------------------
 --------------------------------------------------------------
                     Program Identification                    |
 --------------------------------------------------------------
 --------------------------------------------------------------
                           First CCW                           |
 --------------------------------------------------------------
 --------------------------------------------------------------
                          Failing CCW                          |
 --------------------------------------------------------------
 --------------------------------------------------------------
 Not Used|                     CSW                             |
 --------------------------------------------------------------
 --------------------------------------------------------------
                    Sense Bytes                | Unit Address |
 --------------------------------------------------------------
 --------------------------------------------------------------
                       Last Seek Address                       |
 --------------------------------------------------------------
 --------------------------------------------------------------
                    Volume Label               |   Not Used   |
 --------------------------------------------------------------
 -------------------------------
           Device Type          |
 -------------------------------
|<-------------------------8 bytes------------------------->|
```

Figure 28. Outboard Record Entry Format

Record Entry Label (3 bytes)
contains the record entry ID field of the EBCDIC characters "SER." The label identifies the record as a dynamic error environment output record of OBR, SER0, or SER1.

System Identifier (1 byte)
contains a blank. This field identifies the SER routine (OBR) that wrote the record. The identifier for SER0 is 0, and SER1 is 1.

Record Entry Type (1 byte)
contains an EBCDIC character identifying the type of error that caused the record to be written. The characters are:

CPU machine check (C)
Channel inboard error (I)
I/O outboard error (O)

Date (4 bytes)
contains the current date in packed decimal format (00YYDDDZ) as specified in the CVT.

Time (4 bytes)
contains the current time of day after conversion of the pseudo-clocks to binary format.

Program ID (8 bytes)
contains the program identification as specified in the request block for the program in progress at the time of the error.

First CCW (8 bytes)
contains the first CCW in the user's channel program as specified in the IOB addressed by the request element provided by the IBM-supplied error routine.

Failing CCW (8 bytes)
contains the failing CCW within the channel program as computed from the CSW command address stored in the IOB.

CSW (8 bytes)
contains seven bytes of the CSW (the first byte is not available), stored by the I/O supervisor in the IOB for the I/O request.

Sense Bytes (6 bytes)
contains the sense information stored by the I/O supervisor in the UCB representing the device in error.

Channel and Device Address (2 bytes)
contains the channel and device address for the failing device as specified in the UCB representing that device.

Last Seek Address (8 bytes)
contains the last seek address, as specified in the IOB, for a failing direct access device.

Volume Label (6 bytes)
contains the volume label when the failing device is tape or direct access .

Device Type (4 bytes)
contains the device type field as specified in the UCB for the device in error.

```
    |<----------------------------8 bytes---------------------------->|
-10) ----------------------------------------------------------------- 
    |                                                                 |
    |                           Count Key                             |
    |                                                                 |
   0|-----------------------------------------------------------------|
    |                       |        |   CPU    |          |          |
    |  Record Entry Label   |Sys. I.D.|Model No.| R.e. Type| Not Used |
   8|-----------------------------------------------------------------|
    |                                |                                |
    |              Date              |              Time              |
    |                                |                                |
  16|-----------------------------------------------------------------|
    |                                                                 |
    |                    Program Identification                       |
    |                                                                 |
  24|-----------------------------------------------------------------|
    |                                                                 |
    |                 First CCW of Failing Chain                      |
    |                                                                 |
  32|-----------------------------------------------------------------|
    |                                                                 |
    |                         Failing CCW                             |
    |                                                                 |
  40|-----------------------------------------------------------------|
    |                                                                 |
    |                     Channel Status Word                         |
    |                                                                 |
  48|-----------------------------------------------------------------|
    |                                                                 |
    |                 Channel/Unit Addresses of                       |
  56|                    Active I/O Units at                          |
    |                    the Time of Failure                          |
    |                                                                 |
  64|                                ---------------------------------|
    |                                |                                |
    |                                |   Channel Type Assignment      |
  72|-----------------------------------------------------------------|
    |                                                                 |
    |          Error Recovery Procedure Interface Bytes               |
    |                                                                 |
  80|-----------------------------------------------------------------|
    |               |                 |                               |
    |      CUA      | Channel Status  |                               |
    |               |(for multiprocessing)|                           |
  88|---------------------------------                                |
    |                                            I/O                  |
    |                                          Hardware               |
    |                                           Logout                |
    |                                         (24 bytes)              |
     -----------------------------------------------------------------
```

• Figure 29. Inboard Record Entry Format

Error Recovery Procedure Interface Bytes (8 bytes)

(They are defined in Section V.) When CCH formats the inboard record entry, the ERPIB are inserted in this 8-byte field. When the system is operating in the multiprocessing mode, the first byte of this field within the inboard record entry is redefined to contain multiprocessing information as follows:

Bits 0,1,2 - Unused.

Bit 3 - Multi-System Feature present.

Bit 4 - ID of failing CPU.
If CPU1, bit 4 is zero.
If CPU2, bit 4 is one.

Bit 5 - Unused.

Bits 6,7 - CPU status:
00 - Shared
01 - Partitioned
10 - Shared, but one CPU is reconfigured out.

If the system is not in the multiprocessing mode, byte 0 of the ERPIB field will remain unchanged (containing all zeros).

Channel Status (For Multiprocessing) (2 bytes)
contains the channel status (CHANSTAT) for each CPU if the system is in the multiprocessing mode. The first seven bits of each byte correspond to the seven possible channels for each CPU, and each is set to one if that channel is offline or not included in the system configuration. The last bit of each of the 2 bytes is unused. If not in the multiprocessing mode, CHANSTAT contains all zeros.

Appendix E: Dictionary of Acronyms and Abbreviations

The following acronyms and abbreviations are used in the text, control blocks, and charts of this manual.

Acronym	Meaning
ADCON	address constant
AEQ	asynchronous exit queue
ALU	arithmetic logic unit
APPDG	appendage
ATNTAB	attention table
BTAM	basic teleprocessing access method
CAW	channel address word
CCC	channel control check
CCH	channel check handler
CCK	channel control check
CCW	channel command work
CDC	channel data check
CE	channel end
CRC	cyclic redundancy check
CSW	channel status word
CU	control unit
CUA	channel and unit address
CVT	communications vector table
DCB	data control block
DE	device end
DEB	data extent block
ECB	event control block
EOF	end of file
ERG	erase gap
ERP	error recovery procedure
ERPIB	error recovery procedure interface bytes
EXCP	execute channel program
FIFO	first-in first-out
FLIH	first-level interruption handler
HA	home address
HALTEOD	Halt End-of-Day (operator command)
HDR	header (record)
HIO	Halt I/O (instruction)
ICC	interface control check
ID	identification
IOB	input/output block
LCH	logical channel
LPSW	Load PSW (instruction)
MCH	machine check handler or machine check
MFT	multiprogramming with a fixed number of tasks
MPX	multiplexor
MVT	multiprogramming with a variable number of tasks
M/T	multi-track
NOP	no-operation
OBR	outboard recorder
PCI	program-controlled interruption
PCP	primary control program
PFR	punch feed read
PRT	protection
PSW	program status word
QSAM	queued sequential access method
QTAM	queued teleprocessing access method

Dictionary of Acronyms and Abbreviations (Continued)

Acronym	Meaning
RE	record entry
RB	request block
RQE	request queue element
R0	record zero
R/W	read/write
SDR/CCR	statistical data recorder/channel check recorder
SER	system environment recording
SIO	Start I/O (instruction)
SIRB	system interruption request block
SVC	supervisor call
SYSGEN	system generation

Acronym	Meaning
TAU	tape adapter unit
TCB	task control block
TCC	track condition check
TCH	Test Channel (instruction)
TIC	Transfer in Channel (command)
TIE	Track in Error (command)
TIO	Test I/O (instruction)
TU	tape unit
UCB	unit control block
UCK	unit check
WTO	write-to-operator

Index

IBM®

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

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

READER'S COMMENT FORM

IBM System/360 Operating System
Input/Output Supervisor
Program Logic Manual

Form Y28-6616-4

- Is the material:

	Yes	No
Easy to read?	□	□
Well organized?	□	□
Complete?	□	□
Well illustrated?	□	□
Accurate?	□	□
Suitable for its intended audience?	□	□

- How did you use this publication?

  □ As an introduction to the subject
  □ For additional knowledge
  Other

- Please check the items that describe your position:

  □ Customer personnel
  □ IBM personnel
  □ Manager
  □ Systems Analyst
  □ Operator
  □ Programmer
  □ Customer Engineer
  □ Instructor
  □ Sales Representative
  □ Systems Engineer
  □ Trainee
  Other

- Please check specific criticism(s), give page number(s), and explain below:

  □ Clarification on page(s)
  □ Addition on page(s)
  □ Deletion on page(s)
  □ Error on page(s)

Explanation:

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

YOUR COMMENTS PLEASE . . .

This publication is one of a series which serves as reference for systems analysts, programmers and operators of IBM systems. Your answers to the questions on the back of this form, together with your comments, will help us produce better publications for your use. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Please note: Requests for copies of publications and for assistance in utilizing your IBM system should be directed to your IBM representative or to the IBM sales office serving your locality.

Fold Fold

FIRST CLASS
PERMIT NO. 81
POUGHKEEPSIE, N.Y.

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

POSTAGE WILL BE PAID BY

IBM Corporation
P.O. Box 390
Poughkeepsie, N.Y. 12602

Attention: Programming Systems Publications
Department D58

Fold Fold

IBM®

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

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

IBM Technical Newsletter

File Number S360-30

Re: Form No. Y28-6616-4

This Newsletter No. Y28-2363

Date February 1, 1969

Previous Newsletter Nos. None

IBM SYSTEM/360 OPERATING SYSTEM
INPUT/OUTPUT SUPERVISOR
PROGRAM LOGIC MANUAL

This Technical Newsletter, a part of release 17 of IBM System/360 Operating System, provides replacement pages for Input/Output Supervisor Program Logic Manual, Form Y28-6616-4. These replacement pages remain in effect for subsequent releases unless specifically altered. Pages to be inserted and/or removed are listed below.

65,66

A change to the text or a small change to an illustration is indicated by a vertical line to the left of the change; a changed or added illustration is denoted by the symbol • to the left of the caption.

Summary of Amendments

Changes have been made to correct printing errors.

Note: File this cover letter at the back of the manual to provide a record of changes.

IBM Corporation, Programming Systems Publications, P.O. Box 390, Poughkeepsie, N.Y. 12602

PRINTED IN U.S.A.

When processing is complete, the DEVTYPE routine places a return code in general register 15. An error return code of 04 indicates one of the following conditions:

- No output area specified: the area parameter is missing from the DEVTYPE macro instruction.
- DD name not found: there is no TIOT entry that corresponds to the DD name supplied.
- Invalid UCB unit type field: the UCB unit type field (byte 4 of the device code field) does not specify a direct access, tape, or unit record device.

If the request is completed satisfactorily, the DEVTYPE routine places a return code of 00 in general register 15, and exits via SVC 3.

IOHALT ROUTINE

When the IOHALT routine (SVC 33) receives control, it is given the address of the UCB associated with the device to be stopped. It checks that address for validity, then inspects the UCB device code field to make sure that the device is not a direct access device.

IOHALT branches to a resident I/O Supervisor subroutine to issue the HIO instruction and examine the resulting condition code. If no error occurs, control returns to IOHALT.

The appropriate condition code is placed in general register 15. If the operation was successful, the IOHALT routine places a post code (X'48') in the ECB code field of the IOB and issues SVC 3 to exit.

SECTION V: CONTROL BLOCK AND TABLE FORMATS

The I/O supervisor uses control blocks and tables to communicate with itself, with the rest of the control program, with processing programs, and with I/O devices.

This section of the publication describes in detail the characteristics of the control blocks and of the tables used by the I/O supervisor.

Figures 8 through 21 show control block and table formats. Where pertinent, a discussion of the fields follows the figures.

For convenience, the following control blocks and tables are presented in alphabetical order:

- Attention Table
- Channel Search Table
- Data Control Block
- Data Extent Block
- Device Table
- Error Recovery Procedure Interface Bytes (ERPIB)
- Event Control Block
- Input/Output Block
- Logical Channel Word Table
- Request Element Table
- Statistics Table
- UCB Lookup Table
- Unit Control Block

ATTENTION TABLE

The attention table is used by the I/O supervisor to obtain the addresses of the attention routines required to service the I/O devices attached to the system.

The attention table has the following characteristics:

- Creation. The table is created at system generation time.
- Storage Area. The table resides, as a permanent part of the resident supervisor, in protected resident storage (when protection is available).
- Size. The table contains one 4-byte entry per attention routine, up to a maximum of 64 entries.
- Means of Access. The ATNTAB byte, supplied by the user in the UCB, is added to the starting address of the attention table to obtain the proper attention routine entry in the table for the device.
- Format. The format of an attention table entry is shown in Figure 8.

```
┌─────────────────┬──────────────────────────────────────────────┐
│     Unused      │          Attention Routine Address           │
└─────────────────┴──────────────────────────────────────────────┘
|<----1 byte---->|<--------------------3 bytes--------------------->|
|<-----------------------------4 bytes------------------------------>|
```

Figure 8. Attention Table Entry Format