IBM VSE/Enterprise Systems Architecture
VSE Central Functions

Error Recovery and Recording Transients Diagnosis Reference

Version 6 Release 1

SC33-6326-00

IBM VSE/Enterprise Systems Architecture
VSE Central Functions

Error Recovery and Recording Transients Diagnosis Reference

Version 6 Release 1

SC33-6326-00

Notice!

Before using this information and the product ist supports, be sure to read the general information under “Notices” on page vii.

First Edition (April 1995)

This edition applies to Version 6 Release 1 of IBM VSE/Advanced Functions, which is part of VSE/Central Functions, Program Number 5686-066, and to all subsequent releases and modifications until otherwise indicated in new editions.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the addresses given below.

A form for readers' comments is provided at the back of this publication. If the form has been removed, address your comments to:

IBM Deutschland Entwicklung GmbH
Department 3248
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany

You may also send your comments by FAX or via the Internet:

Internet: s390id@de.ibm.com
FAX (Germany): 07031-16-3456
FAX (other countries): (+49)+7031-16-3456

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, New York 10594, U.S.A.

Any pointers in this publication to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement. IBM accepts no responsibility for the content or use of non-IBM Web sites specifically mentioned in this publication or accessed through an IBM Web site that is mentioned in this publication.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

```
IBM Deutschland Informationssysteme GmbH
Department 0215
Pascal Str. 100
70569 Stuttgart
Germany
```

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks and Service Marks

The following terms, denoted by an asterisk (*) in this publication, are trademarks of the IBM Corporation in certain countries:

```
AT
CUA
IBM
S/370
System/370
System/390
VSE/ESA
VTAM
```

Preface

This Diagnosis Reference documents the logic of the VSE/Central Functions error recovery and recording transients. These are the transients which get loaded in the Physical Transient Area (PTA) and the RAS transient area (RTA).

Manual Organization

The manual consists of the following parts:

- INTRODUCTION to A- and R-transients with both error recovery and error recording functions for errors reported to the system from hardware.
- DESIGN and ORGANIZATION INFORMATION, describing the operation of A- and R-transients, their interface to the supervisor routines and calling relationships of transient phases and phase groups.
- DATA AREAS, showing the formats of error records handled by A- and R-transients and of various control tables used by them.
- DIAGNOSTICS, containing message and phase cross references.
- An index to help finding details fast.

Referenced Publications

The text refers to the following IBM publications:

- *VSE/Central Functions Diagnosis Reference: Supervisor*, SC33-6323-00
- *VSE/Central Functions Diagnosis Reference: Logical Transients*, SC33-6324-00
- *VSE/Central Functions Diagnosis Reference: Initial Program Load and Job Control*, SC33-6325-00
- *VSE/Central Functions Diagnosis Reference: LIOCS, Vol.1*, LY33-9116
- *VSE/ESA 2.1 Messages and Codes*, SC33-6607

Prerequisite Publications

Prerequisite reading for this manual is:

- Function of the operating system as described in *VSE/ESA Guide to System Functions*, SC33-6611-00

Titles and abstracts of other related publications are listed in the *IBM System/370, 30xx and 4300 Processors Bibliography*, GC20-0001.

Introduction

Transients are phases which reside in the system library and get loaded into main memory when they are needed. These perform functions which are needed infrequently and do not have critical performance requirements.

The error recovery and recording transients perform much of the VSE/AF Error Recovery Procedure (ERP) processing, including:

- Recording (recording of Statistical Data Records (SDRs) in the system recorder file, IJSYSRC)
- Issuing operator messages
- Setting up for retry of I/O
- Keeping error statistics
- Causing errors to be reported to I/O requesters

They also process Logging Requests. See *VSE/Central Functions Diagnosis Reference: Supervisor* for a discussion of ERP and recording and how the supervisor uses the error recovery and recording transients to perform ERP and recording.

The error recovery and recording transients are divided into two types:

- The "A-transients" are transients which the ERP system task loads into the Physical Transient Area (PTA) and executes. These transients deal with I/O-related error conditions except the channel control checks and channel interface control checks. A-transients also do recording for Recording Requests (e.g. for SVC 44).
- The "R-transients" are transients which the RAS system task loads into the RAS Transient Area (RTA) and executes. These transients deal with machine checks, channel control checks, interface control checks and channel report words (CRWs).

A-Transients

A-transients do all recording of SDRs (Statistical Data Records, including OBRs, MDRs, etc.) related to I/O devices, and some other recording. They build the SDRs and do the I/O to write them to the recorder file, IJSYSRC. These SDRs serve as input to EREP (Environmental Record Editing and Printing Program), which produces listings in order to analyze the state of the hardware components of the system.

The following types of records are written on IJSYSRC by A-Transients. Each type is identified by a code in the first byte of the error record.

The types of SDRs recorded by A-transients are in the table below. The record type codes are given in hexadecimal.

Code	Record Type	When Recorded
30 30	Device error records Counter overflow records	Recorded due to unit check
34 50 70 80 90 90 91	BTAM–ES records IPL records Missing interrupt handler records (MIH) System EOD records Device EOD records Volume dismount records Misc data records (MDR)	Recorded due to Logging Request

The customer engineer may request specific errors to be recorded, using the MODE command.

Except for disk and cartridge tape devices, A-transients control recovery of errors. This may involve simply retrying I/O, or elaborate procedures to make it appear to an I/O requester that an error condition never arose. This usually involves a lot of cooperation from the supervisor, however.

Finally, A-transients, along with the supervisor, analyze error conditions and decide what to report to the I/O requester.

The ERP system task loads and executes A-transients. It loads them into the Physical Transient Area (PTA), which is used only for A-transients, and holds one transient. The phase names of A-transients begin with "$$ABER".

There are two possible exits from an A-transient: An SVC 5 to load and enter another A-transient or a branch to the routine ERPEXIT in the supervisor. In the latter case, when the A-transient is handling an error condition, the value in Register 15 requests one of 3 types of processing:

Cancel Recognize an unrecoverable I/O error. Post the I/O complete and report the error to the I/O requester and cancel the requesting job if appropriate.

Ignore Ignore the error condition. Post the I/O complete as if no error condition had arisen.

Retry Reschedule I/O. This may be a restart of the original channel program or may be different I/O, but is to be issued as part of the original I/O request.

R-Transients

The RAS system task loads and executes R-transients. It loads them into the RAS Transient Area (RTA), which is used only for R-Transients, and holds one transient. The phase names of R-transients begin with "$$RAST".

ERP Terms

There are many terms used in discussing ERP logic which do not have obvious meanings, but are essential to understanding the discussions in this book:

ERP

This term is derived from "Error Recovery Procedures" and is all of the logic which processes error conditions. ERP is implemented by error recovery and recording transients and the supervisor. Most of ERP is executed by the ERP, DSK, RAS, and SNS system tasks.

error condition

An error condition is a condition that consists of an error that has occurred on a CPU, channel, or I/O device, or a condition that gets reported or is handled like such errors. For example, the occurrence of a machine check is an error condition. Any unit check on an I/O device constitutes an error condition, even though some of them report perfectly normal device operation or serve purely informational purposes.

failing CCW

The failing CCW is the CCW directly associated with an error condition. Not all error conditions have a failing CCW defined for them. In the case of a unit check, the failing CCW is the one which resulted in the channel command which received unit check status. In the case of a device not operational or intercept error condition, the failing CCW is the first CCW in the channel program that was to be started. Note that the existence of a failing CCW does not necessarily mean I/O failed.

I/O request

An I/O request is the request for processing which is represented by a unique CHANQ entry.

IJSYSRC

IJSYSRC is the system recorder file, also known as the error recording data set or ERDS. This file is defined by a regular label definition and the DEFINE IPL statement. IJSYSRC is where all SDRs get recorded.

recording

Recording is the writing of a record to the system recorder file, IJSYSRC, to record an event or accumulated statistical data. Each of these records is called a Statistical Data Record (SDR). Recording is sometimes known as logging.

original error condition

An original error condition is an error condition that is associated with I/O which was not done under ERP control. I/O is done under ERP control only when ERP is processing an error condition.

Recording Request

A request by some task to have the ERP system task record an SDR for a reason other than processing of an error condition. Recording Request processing is one of the two kinds of processing that A-transients do. The other kind is error condition processing. An A-transient may do recording as part of error condition processing, in which case no Recording Request is involved. Recording Requests are made via SVC 44, by the missing interrupt handler, and by the I/O interrupt handler upon request by a BTAM channel end appendage.

SDR

SDR is derived from Statistical Data Record and is the generic term for any record written on IJSYSRC. SDRs are classified into more specific record types, like OBR and MDR.

A-Transients

The ERP system task loads and executes A-transients. It loads them into the Physical Transient Area (PTA), which is used only for A-transients, and holds one transient. The phase names of A-transients begin with "$$ABER".

The A-transients are listed here, grouped by function, with brief descriptions of their functions. This is followed by a complete discussion of the logic and function of the A-transients and how they relate to each other and to the supervisor.

Summary of A-transients

This section summarizes the function of each A-transient in a few sentences. The A-transients are grouped into functional groups. Some of the A-transients fit into more than one group and are covered in more than one section. Table 1 lists the A-transients alphanumerically by name and tells which sections discuss them.

Table 1 (Page 1 of 2). Alphanumeric list of A-transient phases

PHASE	Functional Group
$$ABERAA	“2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)” on page 7
$$ABERAB	“2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)” on page 7
$$ABERAC	“2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)” on page 7
$$ABERAD	“2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)” on page 7
$$ABERAE	“2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)” on page 7
$$ABERAF	“2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)” on page 7
$$ABERAG	“2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)” on page 7
$$ABERAI	“2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)” on page 7
$$ABERAJ	“8809-Type Tape ERP, Recording Request (8809, 9346, 9347)” on page 7
$$ABERAK	“8809-Type Tape ERP, Recording Request (8809, 9346, 9347)” on page 7
$$ABERAL	“8809-Type Tape ERP, Recording Request (8809, 9346, 9347)” on page 7
$$ABERAN	“ERP for 3886, 3890, 3895 devices” on page 8
$$ABERAO	“ERP for 3886, 3890, 3895 devices” on page 8
$$ABERAQ	“ERP for 3886, 3890, 3895 devices” on page 8
$$ABERAR	“ERP for 3886, 3890, 3895 devices” on page 8
$$ABERAS	“ERP for 3886, 3890, 3895 devices” on page 8
$$ABERAT	“ERP for 3886, 3890, 3895 devices” on page 8
$$ABERAY	“3480-Type Tape ERP and Recording Request (3424, 3480, 3490, 3490E, 9348)” on page 7
$$ABERA1	“Recording” on page 8
$$ABERA2	“Recording” on page 8
$$ABERA3	“Recording” on page 8

Table 1 (Page 2 of 2). Alphanumeric list of A-transient phases

PHASE	Functional Group
$$ABERA4	"Recording" on page 8 "2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)" on page 7
$$ABERA5	"Recording" on page 8
$$ABERA6	"Recording" on page 8
$$ABERA7	"Disk ERP and Recording Request" on page 7
$$ABERA9	"Recording" on page 8
$$ABERBA	"ERP for 3800 Printer" on page 8
$$ABERBC	"ERP for 3800 Printer" on page 8
$$ABERBD	"ERP for 3800 Printer" on page 8
$$ABERBE	"Disk ERP and Recording Request" on page 7
$$ABERBF	"Miscellaneous" on page 9
$$ABERD1	"Disk ERP and Recording Request" on page 7
$$ABERES	"Miscellaneous" on page 9
$$ABERJ1	"Miscellaneous" on page 9
$$ABERRA	"Miscellaneous" on page 9
$$ABERRF	"Miscellaneous" on page 9
$$ABERRG	"Miscellaneous" on page 9
$$ABERRJ	"Miscellaneous" on page 9
$$ABERRK	"Miscellaneous" on page 9
$$ABERRP	"Miscellaneous" on page 9
$$ABERRQ	"Miscellaneous" on page 9
$$ABERRS	"Miscellaneous" on page 9
$$ABERRT	"Miscellaneous" on page 9
$$ABERRX	"Miscellaneous" on page 9
$$ABERRY	"Miscellaneous" on page 9
$$ABERR7	"Miscellaneous" on page 9
$$ABERXA	"ERP for Page-Printers" on page 8
$$ABERXB	"ERP for Page-Printers" on page 8
$$ABERXC	"ERP for Page-Printers" on page 8
$$ABERXD	"ERP for Page-Printers" on page 8
$$ABERXE	"ERP for Page-Printers" on page 8
$$ABERXF	"ERP for Page-Printers" on page 8
$$ABERXG	"ERP for Page-Printers" on page 8
$$ABERXH	"ERP for Page-Printers" on page 8

2400-Type Tape ERP (24XX, 3410/11, 3420, 3422, 3430)

$$ABERAA Updates the error statistics in the PUB2 table. First phase to run for an original error condition.

$$ABERAB Handles data check on read and write commands except read opposite recovery. Transfers to other A-transients for other error conditions. Runs immediately after $$ABERAA for original error conditions.

$$ABERAC Handles data checks on erase gap command, errors during repositioning, write data checks after retries fail, and read data checks with bad IBG 0 after retries fail.

$$ABERAD First read backward recovery phase

$$ABERAE Handles all errors except those handled by $$ABERAB, $$ABERAC, and $$ABERAG. Also continues ERP after completion of rewind-unload as part of rewind-unload repositioning.

$$ABERAF Second read backward recovery phase. Also permanent data check handler.

$$ABERAG Handles unsolicited interrupt, intercept condition, finishes ERP after $$ABERAE requests a reposition.

$$ABERAI Sets up for read backward recovery when failing channel program contains IDALs.

$$ABERA4 Builds MDRs and OBRs

Note: This A-transient is also used for PRT1 printers and 35xx card devices.

8809-Type Tape ERP, Recording Request (8809, 9346, 9347)

$$ABERAJ Main ERP. First phase to run for any error condition.

$$ABERAK Builds an MDR as part of Recording Request recording.

$$ABERAL Builds an OBR.

3480-Type Tape ERP and Recording Request (3424, 3480, 3490, 3490E, 9348)

$$ABERAY Recording Request: Re-initialize PUB2 fields, posts requester, returns to supervisor.

3480-Type Tape error recovery A-transients are replaced by a supervisor resident error recovery routine which is discussed in *VSE/Central Functions Diagnosis Reference: Supervisor.*

Disk ERP and Recording Request

$$ABERA7 Builds I/O device SDRs disk devices.

$$ABERBE Builds an OBR or MDR for disk devices in Recording Request recording

$$ABERD1 Selection of message text for disk devices. Passes control on to message writer or recording record builder.

ERP for 3886, 3890, 3895 devices

$$ABERAN Handles 3886 OCR (optical character reader) errors.

$$ABERAO Handles 3895 device errors (phase 1).

$$ABERAQ Handles 3895 device errors (phase 2) and builds I/O device record (type x'30').

$$ABERAR Builds MDR records for 3895 device.

$$ABERAS Handles 3890 device errors (phase 1). Documented in 3890 support manual.

$$ABERAT Handles 3890 device errors (phase 2). Documented in 3890 support manual.

Recording

This section covers recording A-transients that do not fit under any other heading.

$$ABERA1 Writes I/O device SDRs on IJSYSRC.

$$ABERA2 Issues messages regarding the status of IJSYSRC.

$$ABERA3 Writes SDRs on IJSYSRC, if it is on a CKD device, for a Recording Request except for a record-builder Recording Request.

$$ABERA9 Writes SDRs on IJSYSRC, if it is on an FBA device, for Request recording except for a record-builder Recording Request.

$$ABERA4 Builds OBRs for PRT1 printers and 35xx card devices.

Note: This A-transient is also used for tape ERP.

$$ABERA5 Builds I/O device records for BTAM-ES (type x'34').

$$ABERA6 Sets OS/VS device class and type code in OBR (type x'30' to type x'34') and MIH (type x'70') records. Enqueues or dequeues IJSYSRC for enqueue/dequeue Recording Requests.

ERP for 3800 Printer

$$ABERBA Analyzes the error condition and updates the error statistics in the PUB2.

$$ABERBC Builds an MDR record.

$$ABERBD Builds an OBR record.

ERP for Page-Printers

The phases $$ABERXA, $$ABERXB, $$ABERXC, $$ABERXD, $$ABERXE, $$ABERXF, $$ABERXG and $$ABERXH belong to the seperate program product VSE/PSF.

Message Writer

The Message Writer A-transients are replaced by a supervisor resident ERP Message Writer which is discussed in *VSE/Central Functions Diagnosis Reference: Supervisor.*

Whenever the first A-transient $$ABERRL of the old Message Writer is called, control is passed to the resident ERP Message Writer.

Miscellaneous

$$ABERBF 4248 printer ERP.

$$ABERES: Escon director recovery transient.

$$ABERRA: Monitor phase for non-tape devices.

$$ABERRF: Handles PRT1 printer

$$ABERRG: Handles 3505, 3525 card device errors.

$$ABERRJ Old transient formerly used for 23xx disk devices. This A-transient is also used for ERP for many other devices.

$$ABERJ1: Update the error statistics in the PUB2 for card devices.

$$ABERRK: Builds long and short form of OBR records for 3540 diskette I/O unit, 54xx card devices, MICR and OCR devices.

$$ABERRP: Builds MDR records for PRT1 printers (phase 1).

$$ABERRQ: Builds MDR records for PRT1 printers (phase 2).

$$ABERRS: Attempts 1419 MCR (magnetic ink character reader) recovery.

$$ABERRT: Attempts 1287, and 1288 OCR (optical character reader) recovery.

$$ABERRX: Attempts recovery for 1403, 1443, 3203, and 5203 printer, 3881 optical mark reader, 1442, 2501, 2520, 2540, 2560, 2596, and 54xx card devices

$$ABERRY: Recovery for the same devices as $$ABERRX.

$$ABERR7: Attempts 3540 diskette recovery. Determines message code and recovery action.

Interface Between A-Transients and Supervisor

This is a summary of the interface between the A-transients and the supervisor. More information is available in the discussion of ERP in *VSE/Central Functions Diagnosis Reference: Supervisor*

The supervisor passes information to an A-transient in the ERBLOC control block. There are two formats for the ERBLOC:

- "Regular format," which is used when the A-transient is to process an error condition, and
- "Alternate Entry format," which is used when the A-transient is to process a Recording Request.

The format and field definitions for the ERBLOC are summarized in “ERBLOC Area” on page 95.

An A-transient returns information to the supervisor in the ERBLOC and in the PUB2 table.

The supervisor passes to an A-transient the address of the PTA (i.e. the load point of the phase) in Register 11, and the rest of the registers are arbitrary.

An A-transient passes back to the supervisor an action code in Register 15, and the rest of the registers are arbitrary.

Interface Between A-Transients

An A-transient may transfer to another A-transient. There are 4 ways an A-transient can pass information to another A-transient:

- In the ERBLOC control block
- In the PUB2 table
- In the RFTABLE control block
- In the PTA.

Functional Relationship Between A-Transients and Supervisor

ERP or Recording Request processing begins in the supervisor. The supervisor may call an A-transient to do further processing. The division of processing between the supervisor and the A-transients varies. For example, In the case of disk ERP, the supervisor does much more processing itself (and the A-transients do less) than for other ERP. Also, for some Recording Requests, A-transients build the SDRs, while for others, the supervisor builds them. To call an A-transient, the supervisor sets up the ERBLOC, loads the A-transient, and transfers control to it. The A-transient does some processing and may either return to the supervisor or load and transfer to another A-transient. Typically, many A-transients run, transferring to each other, before one finally returns to the supervisor. After the supervisor regains control, it does some final processing, based on information returned from the A-transients.

The A-transients often have not completed their processing for a particular error condition when they return to the supervisor. They return sometimes to have the supervisor retry I/O, do additional I/O, or wait for a future I/O request. For this reason, the A-transients typically keep information in the PUB2 table or in GETVISed storage even after returning control to the supervisor. In this way, the next time they are called by the supervisor, they can continue processing for an error condition for which they began processing on a previous call. In this way, one could think of the A-transients and the supervisor as trading off control until processing for a particular error condition is complete.

Figure 1 on page 11 gives a typical flow of control between supervisor, recording A-transients, and recovery A-transients A-transients.

A-transients do not always run in this order and it is not always the case that all the A-transients in the example run. For example, for some devices, ERP does not keep error statistics. For disk ERP, no error recovery is done by A-transients except recording and message writing. For error recovery on cartridge tape devices and for Recording Requests, only recording A-transients run.

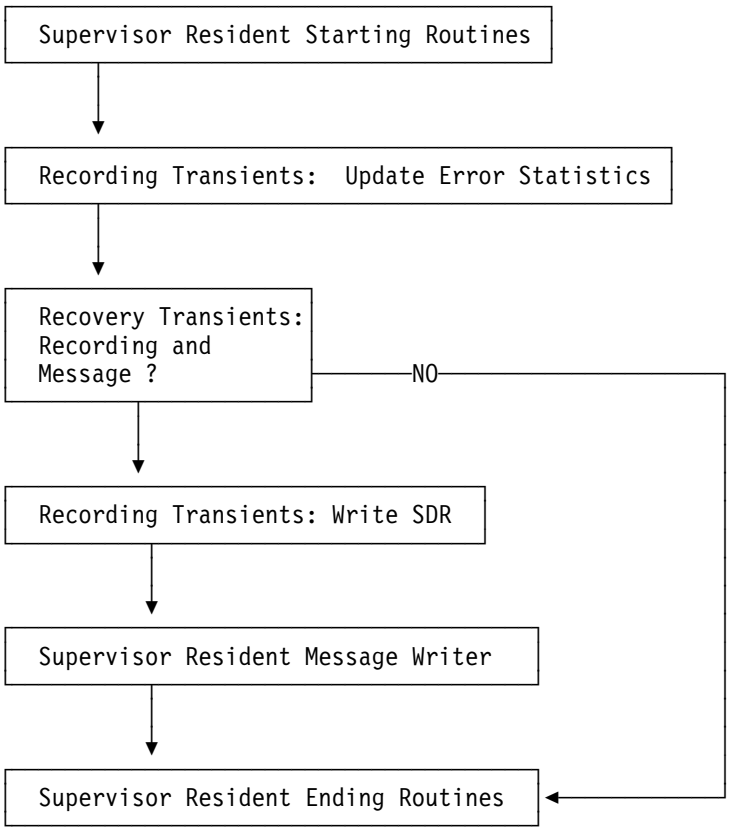

Figure 1. Functional Relationship of A-Transients

A-Transient Entry Conditions

An A-transient's load point must be Byte 0 of the PTA and its entry point must be Byte 10 of the PTA (Bytes 0-9 are used for phase identification information).

At entry to an A-transient, Register 11 must contain the address of the PTA.

Overview of Supervisor Calling of A-Transients

See *VSE/Central Functions Diagnosis Reference: Supervisor* for details on how and why the supervisior calls A-transients. This is a brief overview to assist in understanding of the operation of A-transients.

All A-transients are executed by the ERP system task as part of its processing of an Error Block (in the ERP Error Chain) or a Recording Request. For each Error Block or Recording Request, the supervisor calls one A-transient. That A-transient may transfer to others, and eventually an A-transient returns control to the supervisor. The supervisor then completes processing for the Error Block or Recording Request. During the course of processing a particular error condition, the supervisor may queue and process several error entries (e.g. when retries are involved) for that same error condition.

The A-transient that the supervisor calls is given in Table 2 on page 12 for processing of an error condition and in Table 3 on page 12 for processing of a Recording Request.

Table 2. A-transient supervisor calls for error condition. The conditions are listed in priority order from highest to lowest.

PHASE	Condition under which phase is called
$$ABERRL	ERQPMSG (a flag indicating only messaging is required) is set
$$ABERV1	Device is being used for VTAM* and VTAM is active
$$ABERAJ	Device is 8809, 9346, or 9347 and is not under control of ERP
$$ABERAA	Device is a 2400-type tape device and is not under control of ERP
(P2TNAME)	Device is a non 3480-type tape device and is under control of ERP
$$ABERD1	The device is a disk device
$$ABERRJ	No condition above holds

Note: In the PHASE column, (P2TNAME) means the supervisor calls the A-Transient whose name is given by the P2TNAME field of the PUB2 table entry. While ERP controls the device, the A-transients control P2TNAME.

Table 3. A-transient supervisor calls for Recording Request. The conditions are listed in priority order from highest to lowest.

PHASE	Condition under which phase is called
(parm)	SVC 44 parameters indicate a record-builder request
$$ABERA5	BTAM request (Type 34 SDR)
$$ABERA6	No condition above holds

Note: In the PHASE column, (parm) means the supervisor calls the A-transient named in the SVC 44 parameter list. For record-builder requests only, the SVC 44 parameter list names an A-transient. Only SVC 44 creates a record-builder Recording Request. The A-transient named is as follows, according to the device type of the device for which an SDR is being requested:

Disk	$$ABERBE
3424, 3480, 3490, 9348	$$ABERAY
3800	$$ABERBC or $$ABERBD
8809, 9347	$$ABERAK
3890	$$ABERAT

Details on SVC 44 requests, who makes them, how they are serviced, and how they result in A-transients being called are in *VSE/Central Functions Diagnosis Reference: Supervisor*

Operation of Individual A-Transients

This section groups A-transients into functional groups and explains the function and logic of individual A-transients or groups of A-transients.

Many of the sections contain boxed figures which summarize the function of each A-transient. These figures use the following notation:

- "-------> $$ABERRK" indicates a transfer to A-transient $$ABERRK.
- "RFEXIT=$$ABERRA" near a "------> $$ABERRK" indicates that at the time of the transfer, the A-transient has set the RFEXIT field in the Recorder File Table

to indicate A-transient $$ABERRA. This field may get used later to cause a transfer to the indicated A-transient. Where there is no "RFEXIT=...", the value of RFEXIT is considered irrelevant or is left the same as when the A-transient was entered.

- "------> RETRY" indicates an exit to the supervisor (ERPEXIT), requesting RETRY processing.
- "------> IGNORE" indicates an exit to the supervisor (ERPEXIT), requesting IGNORE processing.
- "------> CANCEL" indicates an exit to the supervisor (ERPEXIT), requesting CANCEL (irrecoverable I/O error) processing.
- "------> (P2TNAME)" indicates a transfer to the A-transient indicated by the P2TNAME field in the PUB2 entry for the concerned device.
- "------> (RFEXIT)" indicates a transfer to the A-transient indicated by the RFEXIT field in the Recorder File Table.

General ERP: $$ABERRJ, $$ABERJ1, $$ABERRA

$$ABERRJ simply transfers to another A-transient unless the device with the error condition is a disk. The A-transient to which $$ABERRJ transfers in the non-disk case is given by the following table:

Table 4. A-transient to which $$ABERRJ transfers (non-disk)

PHASE	DEVICE TYPE
$$ABERRA	2560, 5424, 5425, 3800, adv function printer, 3886, 3895, 3890, 3540, "unsupported" device
$$ABERBF	4248
$$ABERJ1	Everything else (non-disk)

$$ABERJ1 updates statistical counters. If the update causes an overflow, so that an OBR must be recorded, $$ABERJ1 transfers to $$ABERA4 if the device is a 3211 printer and to $$ABERRK otherwise. If there is no overflow, $$ABERJ1 transfers to $$ABERRA.

$$ABERRA is a monitor A-transient for non-tape, non-disk device ERP, except for the 4248 printer. It transfers to another A-transient, choosing one based on device type, status byte, and device status. It may also return to the supervisor if a CCB is not available. See the following sections on specific types of ERP for details on the A-transient to which $$ABERRA transfers.

1270, 1275, 1287, 1288 OCR (Optical Character Reader) Recovery - $$ABERRT

$$ABERRT receives control from $$ABERRA and tests for the following conditions:

CSW bit 44 - Channel Data Check.

- Action: One retry; then takes equipment error exit (retry, cancel).
- Message: 0P28 CHAN DTCHK.

sense byte 0, bit 3 - Equipment Check.

- Action: Posts byte 3 of CCB and then takes continue exit.

Data check and equipment check, which indicate unreadable character and unreadable line, respectively, are retried by LIOCS in an attempt to correct the error.

sense byte 0, bit 1 - Intervention Required.

- Action: Tests for byte 0, bit 6 (nonrecovery) - if present, posts byte 3, bit 4 of the CCB. This indicates that the error is passed back to the application program. Exits via equipment error.
- Messages: 0P35 NON RECOV. If byte 0, bit 6 is not present, take operator intervention exit. 0P08 INTERV REQ.

sense byte 0, bit 6 - Nonrecovery.

- Action: Posts byte 3, bit 4, of CCB and takes continue exit.

sense byte 0, bit 2 - Busout Check.

- Action: One retry; then takes equipment error exit (manual retry, cancel through ATTN routine).
- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 4 - Data Check.

- Action: Posts byte 3, bit 0, of CCB and takes continue exit.

Data check and equipment check, which indicate unreadable character and unreadable line, respectively, are retried by LIOCS in an attempt to correct the error.

sense byte 0, bit 5 - Overrun.

- Action: Four retries; then takes equipment error exit (retry, cancel).
- Message: 0P14 OVERRUN.

sense byte 0, bit 4 - Command Reject.

- Action: Takes program check exit.
- Message: 0P18 COMM REJCT.

CSW bit 47 - Chaining Check.

- Action: Four retries; then takes equipment error exit (retry, cancel).
- Message: 0P14 OVERRUN.

sense byte 0, bit 7 - Keyboard Correction.

- Action: Posts byte 3, bit 1, of CCB and takes continue exit.

sense byte 1, bit 4 - Invalid Font.

- Action: Takes program check exit.
- Message: 0P37 INVLD FONT.

Byte 1, bit 4 applies only to the 1287 in document mode.

14xx, 3203 Models 1 and 2, 5203 Printer Recovery - $$ABERRX, RY

$$ABERRY receives control from $$ABERRA and tests sense information for the following conditions. For the 1443, bits 4 and 5 of sense byte 0 are set to zero.

sense byte 0, bit 7 - Channel 9 Overflow.

- Action: Posts CCB/IORB and exits to continue processing.

This test is resident in real storage.

CSW bit 44 - Channel Data Check.

- Action: If initial selection, one retry. If error persists, exits to the message writer. If initial selection, exits with the allow retry bit on. If channel end, exits with both allow retry and allow ignore bits on.
- Message: 0P28 CHAN DTCHK.

CSW bit 47 - Channel Chaining Check.

- Action: Exits to the message writer with allow retry bit on.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 2 (3203, 5203U), Bit 2, 4, 5 (5203), bit 4, 5 (1403), or Bit 6 (1403U, 1443).

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 3 - Equipment Check.

- Action: Exits to the message writer with the allow ignore bit on.
- Message: 0P10 EQUIP CHK.

sense byte 0, bit 5 - Code General Storage Parity Error (1403U, 3203, 5203U).

- Action: If not channel end, exits to the message writer and then the task is canceled. UCS buffer must be reloaded. Otherwise one retry first.
- Message: 0P33 UCB PARITY.

sense byte 0, bit 1 - Intervention Required.

- Action: If device end without channel end, ignores this condition. Otherwise exits to the message writer for operator intervention.
- Message: 0P08 INTERV REQ.

sense byte 0, bit 2 - Busout Check.

- Action: If initial selection, one retry. If error persists, exits to error recovery message writer. If initial selection , it exits with the allow retry bit on, if channel end, it exits with both allow retry and allow ignore bits on.
- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 4 - Data Check (1403U, 3203, 5203U).

- Action: If byte 0, bit 0 is on, see command reject (next test). Otherwise exits to the message writer with the allow ignore bit on.

- Message: 0P11 DATA CHECK.

sense byte 0, bit 0 - Command Reject.

- Action: If command code is a UCS block data check or reset block data check command, this condition is ignored and processing continues. Otherwise exits to the message writer and then the task is canceled. This procedure allows UCS-oriented programs to operate on non-UCS hardware.
- Message: 0P18 COMM REJCT.

sense byte 0 = 00.

- Action: If not initial selection, exits to the message writer with the allow retry bit on. If initial selection, one retry first.
- Message: 0P19 UNDET ERR.

1255, 1259, 1419 MCR (Magnetic Character Reader) Recovery - $$ABERRS

$$ABERRS receives control from $$ABERRA and tests for the following conditions:

CSW bit 44 - Channel Data Check

- Action: Posts irrecoverable I/O error to CCB/IORB byte 2, bit 2. Turns on PASSBACK bit in the ERQFLG field of the ERBLOC for return to user for device recovery. Exits to the message writer.
- Message: 0P28 CHAN DTCHK.

sense byte 0, bit 0 - Command Reject

- Action: Checks the command code of the CCW causing the interrupt. If the code is hex E1 meaning disengage failed, posts intervention required to CCB byte 3, bit 1; otherwise posts irrecoverable I/O error to CCB byte 2, bit 2. In either case, turns on the PASSBACK bit in the ERQFLG field of the ERBLOC to return to the user for device recovery. Exits to the message writer.
- Messages: 0P18 COMM REJCT and 0P37 DISEN FAIL

sense byte 0, bit 1 - Intervention Required

- Action: Posts intervention required to the CCB (byte 3, bit 1), and exits to supervisor. The error is ignored, no message printed.
- Message: 0P08 INTERV REQ.

The application program should process all documents in the input buffer, note the intervention required, perform any printing necessary for operator recovery, and issue an engage read to the device to continue processing documents. If the intervention required is due to a batch numbering update failure, the operator must update the batch number as part of manual recovery.

sense byte 0, bit 2 - Busout Check

- Action: Posts irrecoverable I/O error to CCB/IORB byte 2, bit 2, and turns on PASSBACK bit in the ERQFLG field of the ERBLOC for return to user for device recovery. Then exits to the message writer.
- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 3 - Should not occur.

- Action: Posts irrecoverable I/O error to the CCB and gives a message to the operator. Exits to the message writer.
- Message: 0P19 UNDET ERR.

CSW bit 47 and sense bits 4, 5, 6, and 7 will not cause an I/O interrupt. If CSW bit 44 or sense bit 1, 2, or 3 is not present for an I/O interrupt, the action and message for sense bit 3 is generated. LIOCS issues two informational messages through the MCR message writer.

1. 4MR1I - EXTERNAL INTERRUPT I/O ERROR
2. 4MR2I - SCU NOT OPERATIONAL

sense byte 0, bit 7 - Batch Numbering Switch Off.

- Action: Posts document buffer byte 0, bits 0 and 1 and inserts reject code hex CF in byte 5. Turns off retry and turns on ignore bits. Exits to the message writer.
- Message: 0P34 NO BATCHNO.

1442 and 2596 Card Device Recovery - $$ABERRX, $$ABERRY

$$ABERRY receives control from $$ABERRA and tests for the following conditions:

CSW bit 44 - Channel Data Check

- Action: If initial selection, one retry. If error persists, exits to the message writer with the allow retry bit on. If data transfer, exits to the message writer for operator intervention.
- Message: 0P28 CHAN DTCHK.

CSW bit 47 - Channel Chaining Check (1442P only)

- Action: Exits to the message writer with the allow retry bit on.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 6.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 3 - Equipment Check.

- Action: Exits to the message writer for operator intervention.
- Message: 0P10 EQUIP CHK.

sense byte 0, bit 1 - Intervention Required

- Action: If device end without channel end, ignores this condition. Otherwise exits to the message writer for operator intervention.
- Message: 0P08 INTERV REQ.

sense byte 0, bit 2 - Busout Check.

- Action: If initial selection, one retry. If error persists, exits to the message writer with the allow retry bit on. If data transfer, exits to the message writer for operator intervention.
- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 4 - Data Check.

- Action: Exits to the message writer for operator intervention.
- Message: 0P11 DATA CHECK.

sense byte 0, bit 5 - Overrun (1442 only)

- Action: Exits to the message writer for operator intervention.
- Message: 0P14 OVERRUN.

sense byte 0, bit 0 - Command Reject.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P18 COMM REJCT.

sense byte 0 = 00.

- Action: Exits to the message writer with the allow retry bit on.
- Message: 0P19 UNDET ERR.

Disk ERP

This section covers the following A-transients:

- $$ABERA7
- $$ABERD1

Recording Request processing for disk devices is covered in “Disk Recording Requests--$$ABERBE” on page 20.

Most of disk ERP is done by the supervisor. The $$ABERD1 transient is only called in two cases. Either the channel status was not zero and it was not a RAS request, or the supervisor resident recovery routine SGDSK requested messaging or recording or both. For the first case, $$ABERD1 selects a message and passes it on to the message writing routine. For the second or third case the only work done by $$ABERD1 is to select the message text and invoke either $$ABERA7 or $$ABERRL.

The operation of the A-transients is shown in Figure 2 on page 19. See the beginning of this section for a description of the notation used in this figure.

```
$$ABERA7:

        If not recording for this device
           (P2NORCM)                      ------>(RFEXIT)

        If supervisor wants an OBR recorded (ERQPOBR)
          Build an OBR
          If supervisor doesn't want a message
             (ERQDMR bit is off; ERQPSOF bit is on
             in ERBLOC)                    ------>$$ABERA6
                                          RFEXIT=IGNORE
          Else                             ------>$$ABERA6
                                          RFEXIT=$$ABERRL
        If supervisor wants an MDR recorded (ERQPMDR)
          Build an MDR
          If supervisor doesn't want a message
             (ERQDMR bit is off; ERQPSOF bit is on
             in ERBLOC)                    ------>$$ABERA1
                                          RFEXIT=IGNORE
          Else                             ------>$$ABERA1
                                          RFEXIT=$$ABERRL
```

Figure 2 (Part 1 of 2). Disk ERP A-Transient Function

```
$$ABERD1:
        If device is not supported
           set MSG 0P20                   ------>$$ABERRL
        If not operational or
           channel prtection check or
           channel program check or
           channel chaining check or
           channel data check
            set message and               ------>$$ABERRL
        If unit check
          set indication for action and
          target of message 0P..x x
          If recording but no messaging
             RFEXIT = IGNORE
             leave to                  ------>$$ABERA7
          determine message number
          If recording and messaging
             RFEXIT = $$ABERRL
             leave to                  ------>$$ABERA7
          If messaging required            ------>$$ABERRL
```

Figure 2 (Part 2 of 2). Disk ERP A-Transient Function

Disk ERP Notes:

1. $$ABERA1 writes the record to IJSYSRC and exits as determined by the RFEXIT field in the Recorder File Table. See “Recording - $$ABERA1, $$ABERA2, $$ABERA6” on page 57.

Disk Recording Requests--$$ABERBE

$$ABERBE builds an OBR or MDR (depending on device type) and transfers to $$ABERA1 to record the SDR. It sets the RFEXIT field in the RFTABLE to indicate itself ($$ABERBE) so that it gets invoked again after the recording takes place. $$ABERBE maintains a flag so that it knows whether it is being invoked intially or after recording. After recording, it posts an ECB in case a task is waiting for the recording.

A more detailed explanation of the function of $$ABERBE is in the following figure:

```
$$ABERBE:
        If re-entered after recording          ------>IGNORE
        If no recording should be done because of
          INTREC bit in SYSCOM, P2NORCM bit in PUB2,
          or IJSYSRC is not ready              ------>IGNORE
        If 2311 or 2314 then
          Build an OBR
          Set P2DMOD field in PUB2             ------>$$ABERA1
                                          RFEXIT=$$ABERBE
        Else
          Build an MDR
          Zero usage count in PUB2 (P2USAGE)
          Clear SDRDISK area in PUB2           ------>$$ABERA1
                                          RFEXIT=$$ABERBE
```

Figure 3. Disk Recording Request A-Transient Function

Notes:

1. Whenever $$ABERBE exits to the supervisor, it sets the traffic bit in the ECB for the Recording Request.
2. $$ABERBE sets the MDR device type code with the code it gets with EXTRACT. If EXTRACT does not successfully return an MDR device type code, $$ABERBE computes one from the PUB device type code (PUBDEVTY). $$ABERBE computes the OBR device type code from PUBDEVTY.

24xx, 3410, 3411, 3420, 3422, 3430 Tape Device ERP

This section covers the following A-transients:

- $$ABERAA
- $$ABERAB
- $$ABERAC
- $$ABERAD
- $$ABERAE
- $$ABERAF
- $$ABERAG
- $$ABERAI

VSE ERP Function

This section describes the ERP function of VSE for 24XX-type devices. This section is somewhat oversimplified in that there are many odd conditions which could cause ERP function to be different from what is given here (e.g. if the recorder file is full, VSE will not record an SDR, even though this section says it will).

The A-transients are largely but not entirely responsible for the function described here.

Equipment Check (Sense Byte 0, Bit 3)

- Action: Issue message and recognize irrecoverable I/O error.
- Message: 0P10 EQUIP CHK.

Busout Check (Sense Byte 0, Bit 2)

- Action: If unit check was in initial status (no Device End), retry. Otherwise: If error was on a write operation, reposition the tape and reissue the command. If on any other command, reissue the command. Follow this procedure 5 times. if recovery fails, issue message and recognize irrecoverable I/O error.
- Message: 0P09 BUSOUT CHK.

Intervention Required (Sense Byte 0, Bit 1)

- No Device End in unit status:
  - Action: Check TU Status B (Sense Byte 1, Bit 2).

    Off: The device is non-existent. Do not attempt retries.

    On: Issue message, retry after unit is readied.
  - Message: 0P08 INTERV REQ. or 0P47 UNX INTERV. (for tapes that have been opened).
- Device End in unit status:
  - Action: If the command was a Rewind Unload, issue message and recognize successful I/O. Put device in Intervention Required status.
  - Message: 0P19 UNDET ERR.

Command Reject (Sense Byte 0, Bit 0)

If at load point and file protect is on:

- Action: If at load point and file protect, issue message and rewind-unload the volume.
- Message: 0P17 FILE PROT.

Else:

- Action: Issue message and recognize irrecoverable I/O error.
- Message: 0P18 COMM REJCT.

Overrun (Sense Byte 0, Bit 5)

- Action: Reposition tape and retry. Follow this procedure 5 times. If recovery fails, issue error message and recognize irrecoverable I/O error.

- Message: 0P14 OVERRUN.

Note that a data check during overrun suppresses the overrun condition.

Load Point (Sense Byte 1, Bit 4)

- If caused by Read Backward operation:
  - Action: Set Unit Exception on and Unit Check off in CCB. Recognize successful I/O.
  - Message: none
- Else:
  - Action: If not caused by read backward operation, issue error message and recognize irrecoverable I/O error.
  - Message: 0P29 AT LOADPT

Data Check (Sense Byte 0, Bit 4)

- Action:

  If the operation is read or read backward, and if the noise bit (Sense Byte 1, Bit 0) is on, or if more than 11 bytes were read, VSE repositions the tape and retries the Read or Read Backward. If the noise bit is on and the blocksize is less than the minimum requirement, VSE recognizes an irrecoverable data check I/O error. If the noise bit is off and the blocksize is less than 12 bytes, VSE simply retries, reading the next block on the tape. VSE follows this procedure 40 times. If the 4th reposition has a Backward Into Loadpoint error, VSE assumes the user is reading the first block on the tape and that the first Interblock Gap (IBG) is bad. In this case, VSE unloads the tape, issues the message 0P39 BAD VOL, and retries after the unit is loaded and readied. VSE follows every fourth retry with a tape cleaner action. VSE chains a Track In Error (TIE) command before every reread to send track-in-error information (sense byte 2) to the tape control unit.

  If the above procedure does not help VSE tries to recover by reading in the opposite direction on the tape. This is called Read Opposite Recovery (ROR). If any of the following conditions exist, however, VSE does not attempt Read Opposite Recovery when:

  - 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 read backward recovery, VSE first routine first issues a read or read backward command as its first retry. On subsequent retries, VSE repositions the tape, issues a Track In Error (TIE) command, and then a Read or Read Backward command. After every fourth retry, VSE performs a tape-cleaner action. VSE retries 40 times. The Read or Read Backward CCW has the Suppress Data Transfer bit set until the first successful retry. Subsequent Read or Read Backward CCWs have the Suppress Data Transfer bit off so they can transfer data. If a read is successful, the routine issues a Forward Space Block or Backward Space Block command to reposition the tape past the block read.

  If all 40 opposite read attempts fail, VSE makes one final retry, this time attempting to read in the original direction. Note that VSE avoids the final read

attempt if either a permanent Busout check occurs when the repositioning command is issued or an Equipment Check occurs.

If VSE was finally unable to read the block: If the user will not accept the data check, VSE issues a message and recognizes and irrecoverable I/O error. If the user will accept data checks, VSE sets the data check bit in the CCB and recognizes a successful I/O completion.

If the operation is a write or write tapemark, VSE repositions the tape, issues an erase gap command, and retries. VSE follows the procedure 15 times. If the 4th reposition has a Backward Into Loadpoint error, VSE assumes the user is trying to write the first block on the tape and that it is not correctly writing the first Interblock Gap (IBG). In this case, VSE unloads the tape, issues the message 0P39 BAD VOL, and retries after the unit is loaded and readied. For all but 24XX tape units, VSE issues a Loop Write to Read command between the 14th and 15th retries. If the retries fail, VSE issues a message recognizes an irrecoverable I/O error. If the operation is an Erase Gap, VSE reissues the command up to 3 times. If it still fails, VSE issues an error message and recognizes an irrecoverable I/O error.

- Message: 0P11 DATA CHECK.

Channel Data Check (CSW bit 44)

- Action: If this condition occurred on a read or write operation, reposition the tape and reissue the command. If it occurred on any other command, reissue the command. Follow this procedure 5 times. If unsuccessful, issue a message and recognize an irrecoverable I/O error.
- Message: 0P28 CHAN DTCHK.

Data Converter Check (Sense Byte 0, Bit 7)

- Action: Issue an error message and recognize an irrecoverable I/O error.
- Message: 0P30 CONVRT CHK.

Not Capable (Sense Byte 1, Bit 7)

- Action: Unload the tape. Retry after operator reloads and readies the device.
- Message: 0P32 NON COMPAT.

Channel Chaining Check (CSW Bit 47)

- Action: Reposition the tape and retry. Follow this procedure 5 times. If unsuccessful, issue a message and recognize and irrecoverable I/O error.
- Message: 0P14 OVERRUN.

PE ID Burst Check (Sense Byte 5, Bit 3)

- Action: Rewind the tape and retry. Follow this procedure 15 times. If unsuccessful, issue a message and unload the tape. Retry after the operator reloads and readies the device.
- Message: 0P44 ID CHECK.

Data Security Erase Failed (Sense Byte 7, Bit 4)

- Action: Issue a message and recognize an irrecoverable I/O error.
- Message: 0P42 DSE FAILED.

File Protect (Sense Byte 1, Bit 6)

- Action: Issue a message and unload the tape. Retry after the operator reloads and readies the tape device.
- Message: 0P17 FILE PROT.

None of the above.

- Action: Issue a message and recognize an irrecoverable I/O error.
- Message: 0P19 UNDET ERR.

A-Transient Operation

The operation of the A-transients is shown in Figure 2 on page 19. See the beginning of this section for a description of the notation used in this figure. In addition to the notation described there, this figure uses the following notation:

- "P2TNAME=$$ABERAB" near a "------> RETRY" indicates that at the time of the transfer, the A-transient has set the P2TNAME field in the PUB2 entry for the relevant device to indicate A-transient $$ABERRA. The supervisor may later call the transient that P2TNAME indicates. Where there is not "P2TNAME=...", the value of P2TNAME is considered irrelevant or is left the same as when the A-transient was entered.
- Next to every "------> RETRY" is an indication of the type of retry that the A-transient requests. Fields in the PUB2 entry determine the type of request. The types in this figure are:

  USER Retry the user channel program (possibly with TIE, LWR, or NOP commands prefixed, possibly from a point after the beginning of the channel program).

  SPACE Do a backspace or forward space I/O to reposition tape for a

  CLEANER Do a backspace or forward space I/O as part of a cleaner action. retry.

  REW Do a rewind to reposition tape for a retry.

  RUN Do a rewind-unload so operator can load a new tape.

  ERG Do an Erase Gap to try to move tape over an unwritable spot.

- Next to a "------>IGNORE", "INTERCPT" means the A-transient requests that the supervisor intercept the next I/O request to the device. This means that the supervisor will assume an error condition on the next I/O request to the device without actually even trying to perform the I/O. It will call A-transients as for other error conditions, with fields in the ERBLOC indicating that the error condition is an intercept condition. See *VSE/Central Functions Diagnosis Reference: Supervisor* for details.

```
$$ABERAA:   (first phase to run for original error)
        If 3430 forced logging error condition
          that hasn't been recorded yet          ------>$$ABERA4
                                                  RFEXIT=$$ABERAA
        Update statistical counters in PUB2
        If threshold reached
          Set RFEVA bit in Recorder File Table to
              indicate an Error Volume Analysis message
              is required.
        If recording is required (Note 1)        ------>$$ABERA4
                                                 RFEXIT=(P2TNAME)
        If RFEVA bit is set                        ------>$$ABERA2
                                                 RFEXIT=(P2TNAME)
        Otherwise                                  ------>$$ABERAB
```

Figure 4 (Part 1 of 9). 24XX-type Tape ERP A-Transient Operation

```
$$ABERAB:
        If Sense command failed (Note 2)         ------>$$ABERAF
        If requester has been posted             ------>$$ABERAG
        If error condition is intercept condition------>$$ABERAG
        If error cond is unsolicited interrupt   ------>$$ABERAG
        If error cond is not operational         ------>$$ABERAE
        Compute address of failing command CCW
        If can't compute CCW addr
          Select 0P20 (Recovery Error) message   ------>$$ABERAF
        If user allows CCB CCW address to be updated
          Update CCBCCW field in CCB
        If failed I/O was ERP's LWR I/O          ------>RETRY
                                                 USER
        If failed I/O was ERP reposition I/O
          If error is Unit Check, Channel Data Check,
              Program Check, Chaining Check, or
              Protection Check                   ------>$$ABERAC
          Else
            * Assume the failed I/O was actually
            *     successful reposition I/O
            If original error was on Write cmd   ------>RETRY
                                                 ERG
            If failed I/O was Rewind             ------>RETRY
                                                 USER
            If ERP is not doing cleaner action or
               cleaner action is now finished    ------>RETRY
                                                 USER
            Otherwise                            ------>RETRY
                                                 CLEANER
```

Figure 4 (Part 2 of 9). 24XX-type Tape ERP A-Transient Operation

```
$$ABERAB, continued:
        If failed I/O was *not* ERP reposition I/O
          If failed I/O was ERP's ERG I/O        ------>$$ABERAC
          If 14 write retries have taken place   ------>$$ABERAC
          If error condition is Unit Check, Channel
             Data Check, Program Check, Chaining Check,
             or Protection Check
            If error is Command Reject, Intervention
               Required, Busout, Backward into
               Loadpoint, Equipment Check, or Overrun
               and $$ABERAE has not yet processed
               it                              ------>$$ABERAE
            If error is data check
              If Write data check
                If 15 Write retries taken place  ------>$$ABERAC
                If failed I/O was ERG            ------>$$ABERAC
                Else                             ------>RETRY
                                                        SPACE
              If read data check
                If error is SAGC check and Fewer than 5
                  read retries have taken place  ------>RETRY
                                                        REWIND
                If error is noise record read    ------>RETRY
                                                        USER
                If fewer than 12 bytes read      ------>$$ABERAF
                If 40 retries have taken place   ------>$$ABERAI
                If # of retries is mult of 4     ------>RETRY
                                                        CLEANER
                Else                             ------>RETRY
                                                        SPACE
            If error is PE ID burst check        ------>$$ABERAC
            Otherwise                            ------>$$ABERAE
          Else (error condition is not Unit Check,
             Channel Data Check, Program Check,
             Chaining Check, or Protection Check)
            If failed I/O was done under ERP
               control
              If Original I/O had Unit Exception and
                 retrying lost it
                Set Unit Exception in CCB        ------>IGNORE
              Else                               ------>$$ABERAE
```

Figure 4 (Part 3 of 9). 24XX-type Tape ERP A-Transient Operation

```
$$ABERAC:
        If failed I/O was ERP's LWR command
          If error is Unit Check              ------>$$ABERA4
                                            RFEXIT=$$ABERAB
          Else                               ------>$$ABERAB
        If failed I/O is ERP reposition I/O
          If error is Backward Into Loadpoint on 4th
             BSB reposition                   ----->RETRY
                                              RUN
          If error is Equipment Check
            Select 0P10 (Equipment Check) msg    ------>$$ABERA4
          If error is Intervention Required without
             Device End or Command Reject
            Select 0P20 (Recovery Error) msg     ------>$$ABERA4
          If error is Busout or Overrun
            HANDLE_REPO_CHANNEL_CHECK (see below)
          If error is Backward Into Loadpoint on a
             cleaner action reposition I/O
            If failed I/O was 1st BSB of cleaner ------>RETRY
                                              RUN
            Else                             ------>RETRY
                                              CLEANER
          If error is data check
            If failed I/O is ERP's ERG command
              If 3 ERG retries have been done
                Select 0P20 (Recovery Error) msg ------>$$ABERA4
              Else                           ------>RETRY
                                              ERG
          If error is Channel Data Check
            HANDLE_REPO_CHANNEL_CHECK (see below)
          If error is any other Unit Check
            Select 0P20 (Recovery Error) msg     ------>$$ABERA4
          If error is Chaining Check
            HANDLE_REPO_CHANNEL_CHECK
```

Figure 4 (Part 4 of 9). 24XX-type Tape ERP A-Transient Operation

```
$$ABERAC, continued
        If failed I/O is *not* ERP reposition I/O
          If 14 write retries and ERP's ERG
             operation have been done
            If device is 24XX                  ------>RETRY
                                                 USER
            If original failed I/O is WTM       ------>RETRY
                                                 USER
          If PE ID Burst Check
            If 15 write retries have been done   ------>RETRY
                                                 USER
                                             P2TNAME=$$ABERAE
            Else                               ------>RETRY
                                                 REWIND
          If 15 write retries have been done
            Select 0P11 (Data Check) message     ------>$$ABERA4
          If failed I/O is an ERG command
            If 3 write retries have been done
              Select 0P11 (Data Check) message   ------>$$ABERA4
            Else                               ------>RETRY
                                                 USER
          Else
            Select 0P20 (Recovery Error) message ------>$$ABERA4


HANDLE_REPO_CHANNEL_CHECK:  (referenced above)

        If reposition channel check count = 5
          Select 0P20 (Recovery Error) message   ------>$$ABERA4
        Else
          If error is Busout on a NOP which is not
            command chained to anything          ------>RETRY
                                                 USER
          Else                                 ------>$$ABERAB
```

Figure 4 (Part 5 of 9). 24XX-type Tape ERP A-Transient Operation

```
$$ABERAE:
        Select an action based on the error condition
            (in priority order):
        Not Operational:
          Select 0P31 (Not Operational) message  ------>$$ABERA4
        3430 Deferred Unit Check:                ------>RETRY
                                                        USER
        Microprogram Error (detected by device):
          Select 0P10 (Equipment Check) message  ------>$$ABERA4
        Equipment Check:
          Select 0P10 (Equipment Check) message  ------>$$ABERA4
        Busout:
          Select 0P09 (Busout Check) message
          HANDLE_CHANNEL_ERROR (see below)
        True Intervention Required:
          Selection 0P31 (not operational), 0P43
              (unexpected intervention), or 0P08
              (intervention required) message,
              depending on exact nature of error ------>$$ABERA4
        Normal Rewind Unload Completion:
          If failed I/O was ERP's Rewind Unload
            Select message that ERP determined before
                requesting the Rewind Unload     ------>$$ABERA4
          Else                                  ------>IGNORE
        File Protected:                          ------>RETRY
                                                        RUN
        other Command Reject:
          Select 0P18 (Command Reject)           ------>$$ABERA4
        Overrun:
          Select 0P14 (Overrun) message
          HANDLE_CHANNEL_ERROR (see below)
        Backward Into Loadpoint on a Read command------>IGNORE
        other Backward Into Loadpoint:
          Select 0P29 (Backward Into Loadpt) msg ------>$$ABERA4
        Data Check:                              ------>$$ABERAB
        Data Security Erase Failure:
          Select 0P42 (DSE Failure) message      ------>$$ABERA4
        Channel Data Check:
          Select 0P28 (Channel Data Check) msg   ------>$$ABERA4
        Data Conversion Check:
          Select 0P30 (Conversion Error) message ------>$$ABERA4
        Not Capable:                             ------>RETRY
                                                        RUN
```

Figure 4 (Part 6 of 9). 24XX-type Tape ERP A-Transient Operation

```
$$ABERAE, continued
        Select an action based on the error condition
            (in priority order), continued:
        PE ID Burst check:                      ------>$$ABERAC
        Chaining Check:
          Select 0P09 (Busout) message          ------>$$ABERA4
          HANDLE_CHANNEL_ERROR                        ------>$$ABERA4
        Channel Program Check:
          Select 0P24 (Program Check)           ------>$$ABERA4
        Channel Protection Check:
          Select 0P25 (Protection Check)        ------>$$ABERA4
        Otherwise:
          Select 0P19 (Undetermined Error) msg  ------>$$ABERA4

HANDLE_CHANNEL_ERROR:
        If 5 channel error retries have been done------>$$ABERA4
        Else
          If error is Busout with Device End on a
             Write command                      ------>RETRY
                                                SPACE
          If error is on Track In Error command
            If TIE cmd chained to something     ------>$$ABERA4
            Else                            ------>RETRY
                                                USER
          Else
            If error is Busout              ------>RETRY
                                                USER
            Else                            ------>RETRY
                                                SPACE
```

Figure 4 (Part 7 of 9). 24XX-type Tape ERP A-Transient Operation

```
$$ABERAG:
        If I/O was done under ERP control
          If failed I/O was a reposition I/O other than
              a cleaner action
            Select 0P20 (Recovery Error) message  ------>$$ABERA4
          Act according to error condition:
            Equipment Check:                    ------>$$ABERAE
            Busout:                            ------>$$ABERAE
            Intervention Required:             ------>$$ABERAE
            Command Reject:                     ------>$$ABERAE
            Overrun:                           ------>$$ABERAE
            Backward Into Loadpoint:            ------>$$ABERAE
            Data Check:
              If doing Read Opposite Recovery     ------>$$ABERAD
              Else                              ------>$$ABERAB
            PE ID Burst Check:                   ------>$$ABERAC
            Other Unit Check:                  ------>$$ABERAE
            Channel Data Check, Protection Check,
                Chaining Check, or Program Check:
              If failed I/O is reposition I/O
                Select 0P20 (Recovery Error) msg  ------>$$ABERA4
              Else                              ------>$$ABERAE
            Otherwise:
              If failed I/O is reposition I/O     ------>RETRY
                                                   USER
                                               P2TNAME=$$ABERAG
              Else                               ------>IGNORE
```

Figure 4 (Part 8 of 9). 24XX-type Tape ERP A-Transient Operation

```
$$ABERAG, continued:
        If I/O was *not* done under ERP control
          If error condition is unsolicited interrupt
            Select 0P43 (Tape Volume Change) msg------>$$ABERRL
          If error condition is intercept condition
            If user accepts unrecoverable I/O errors
                and message selected when intercept was
                requested is 0P46 (Lost Posit'g)------>RETRY
                                                   USER
            Else                              ------>$$ABERA4
          If error condition is Data Security Erase
              Failure                         ------>IGNORE
                                                   INTERCPT
          If error condition is normal Rewind Unload
              Completion                       ------>IGNORE
          Else                                ------>IGNORE
                                                   INTERCPT
```

Figure 4 (Part 9 of 9). 24XX-type Tape ERP A-Transient Operation

Notes:

1. $$ABERAA determines that recording is necessary when one of the following conditions is true:
   - The device is in diagnostic mode, the limit count is not reached, and the CE has specified this mode in the MODE command. must be recorded.
   - The device is in intensive mode, the error is initial, the limit count is not reached, and the CE has specified this mode in the MODE command.
   - A counter in the PUB2 table is full.

   In the first two cases, $$ABERAA sets the RECDERR bit in the Recorder File Table and the limit count is reduced by one. In the third case, $$ABERAA sets the RECDSF bit.

   If the device is in intensive or diagnostic mode and the limit count has been reached (remaining count is 0), $$ABERAA resets the device to normal mode. $$ABERAA does this not at the time the remaining count reaches 0, but at the time of the next error condition after that.
2. The A-transients determine whether the Sense command (which the supervisor always issues following a unit check) failed as follows: If the ERQMSG field upon entry to the first A-transient indicates 0P20 (Recovery Error), the sense command failed. Otherwise, either the Sense command succeeded or there was no Sense command (because there was no unit check).
3. Except where noted otherwise, the P2TNAME field in the PUB2 entry indicates A-transient $$ABERAB upon exit from an A-transient. This is either because the A-transient explicitly set it or because the supervisor initialized it to that value and no A-transient has modified it.

8809, 9346, 9347 Tape Device ERP/Recording Request

This section covers the following A-transients:

- $$ABERAJ
- $$ABERAK
- $$ABERAL

VSE ERP Function

This section describes the ERP function of VSE for 8809-type devices. This section is somewhat oversimplified in that there are many odd conditions which could cause ERP function to be different from what is given here (e.g. if the recorder file is full, VSE will not record an SDR, even though this section says it will).

The A-transients are largely but not entirely responsible for the function described here.

VSE ERP does not keep error statistics for the 8809, 9347, and 9346 tape devices. The 8809 and 9347 keep their own error statistics (called environmental data) and transmit them occasionally to VSE, via unit checks, sense commands, and Read Buffered Log commands, to be recorded. There are no statistics kept or recorded for 9346.

Because the hardware in these devices does retries and other error recovery, any error that is reported to VSE is probably not recoverable, so VSE does not attempt

any recovery. ERP for these devices consists only of recording errors, recording statistics, writing operator messages, and return information to I/O requesters.

The sense data from these devices is mostly compatible with the 24XX, etc. tape devices in Bytes 0-2 and 4-7. Sense Byte 3 is defined uniquely for these devices and most ERP decisions are based on Byte 3.

The VSE ERP function that these A-transients help to implement is described below, arranged by error condition type. Following that is a discussion of how the individual A-transients operate.

If the I/O request has already been posted when an error condition arises, VSE simply recognizes successful I/O completion. This means nothing gets recorded, no message gets written, and no I/O request is affected. If the I/O request has not been posted, the following function applies.

In the following error condition descriptions, the relevant Sense Byte 3 value is given in parentheses, in hexadecimal, if the error condition is a unit check. If there is no value in parentheses, the error condition is not a unit check.

Environmental Data Present (01)

- Action: Record an MDR containing sense data. Recognize successful I/O completion.
- Message: none

Temporary Equip Check Control Overflow (02)

- Action: Record an OBR (except 9346). Retry.
- Message: none

Permanent Equipment Check (03)

- Action: Record an OBR (except 9346), issue message, recognize irrecoverable I/O error.
- Message: 0P10 EQUIP CHK.

Permanent Equipment Check (04)

- Action: Record an OBR (except 9346), issue message, recognize irrecoverable I/O error.
- Message: 0P10 EQUIP CHK.

Intervention Required (06)

- Action: If the unit check is in ending status (which indicates successful completion of a Rewind Unload command), recognize successful I/O completion. Else, issue a message, then retry after the operator readies the device.
- Message: 0P08 INTERV REQ or 0P47 UNX INTERV. (if volume has been opened).

Write Data Check (07)

- Action: Issue message. Record an OBR (except 9346). Recognize irrecoverable I/O error.
- Message: 0P11 DATA CHECK.

Permanent Equipment Check (08)

- Action: Record an OBR (except 9346), issue message, recognize irrecoverable I/O error.
- Message: 0P10 EQUIP CHK.

Read Data Check (09)

- Action: Record an OBR (Except 9346). Issue a message. If user accepts data checks, recognize successful I/O (but set data check bit in CCB). Else, recognize irrecoverable I/O error.
- Message: 0P11 DATA CHECK.

Not Capable (0A)

- Action: Unload the volume. Issue a message. After operator has reloaded and readied the device, retry.
- Message: 0P32 NON COMPAT.

File Protect (0B)

- Action: Unload the volume. Issue a message. After operator has reloaded and readied the device, retry.
- Message: 0P17 FILE PROT.

Command Reject (0C)

- Action: Issue message. Recognize irrecoverable I/O error.
- Message: 0P18 COMM REJCT.

Backward Into Load Point (0D)

- Action: If the failed command is a backspace command-chained to a Read command, consider it a simulation of a Read Backward command, so set Unit Exception instead of Unit Check in the CCB and recognize successful I/O completion. Else, issue a message and recognize an irrecoverable I/O error.
- Message: 0P29 AT LOADPT

Data Security Erase Failed (0E)

- Action: Record an OBR (except 9346). Issue a message. Recognize an irrecoverable I/O error.
- Message: 0P42 DSE FAILED.

PE ID Burst Check (0F)

- Action: Unload the volume. Issue a message. After operator has reloaded and readied the device, retry. Record an OBR (except 9346).
- Message: 0P44 ID CHECK

Overrun (10)

- Action: Issue a message. Record an OBR (except 9346). Recognize an irrecoverable I/O error.
- Message: 0P14 OVERRUN.

Manual Unload (11)

- Action: Issue a message. Recognize an irrecoverable I/O I/O error.
- Message: 0P43 TAPEVOL CH

Unsolicited Device End - Unit Check Interrupt

- Action: Issue a message. Record an OBR. Recognize an irrecoverable I/O I/O error.
- Message: 0P43 TAPEVOL CH

Unsolicited Not Ready to Ready Interrupt Received

- Action: Issue a message. Recognize an irrecoverable I/O error.
- Message: 0P43 TAPEVOL CH

Not Operational

- Action: Issue a message. Recognize an irrecoverable I/O error.
- Message: 0P31 NOT OP

Channel Program Check

- Action: Issue a message. Recognize an irrecoverable I/O error.
- Message: 0P24 PROG CHECK

Channel Protection Check

- Action: Issue a message. Recognize an irrecoverable I/O error.
- Message: 0P25 PROT CHECK

A-Transient Operation

Figure 5 on page 36 gives an overview of the operation of the individual A-transients. The notation in this figure is the same as in Figure 4 on page 25.

```
$$ABERAJ:   (first phase to run for error condition)
        If failed I/O is Rewind Unload reposition
          If original error was PE ID check       ------>$$ABERRL
          Else
            RECORD_AND_MESSAGE (see below)
        If I/O request has been posted            ------>IGNORE
        If error condition is Unit Check
          Act according to the error condition:
          Equipment Check:
            Select 0P10 (Equipment Check) message
            RECORD_AND_MESSAGE (see below)
          Intervention Required:
            If status has DE (final status from Rewind
               Unload)                            ------>IGNORE
            Else
              If volume is open
                Select 0P47 (Unexpectd Interv) msg------>RETRY
                                                   USER
              Else
                Select 0P08 (Interv Reqd) msg     ------>RETRY
                                                   USER
          Commmand Reject:
            Select 0P18 (Command Reject) message  ------>$$ABERRL
          Overrun:
            Select 0P14 (Overrun) message
            RECORD_AND_MESSAGE (see below)
          Backward Into Loadpoint:
            If Failed I/O is BSB chained to READ
              Set UE instead of UC in CCB         ------>IGNORE
            Else
              Select 0P29 (At Loadpoint) message  ------>$$ABERRL
          Read Data Check:
            Select 0P11 (Data Check) message
            RECORD_AND_MESSAGE (see below)
          Write Data Check:
            Select 0P11 (Data Check) message
            RECORD_AND_MESSAGE (see below)
          DSE Failure:
            Select 0P42 (DSE Failed) message
            RECORD_AND_MESSAGE (see below)
          Not Capable:
            Select 0P32 (Non Compatible) message  ------>RETRY
                                                   RUN
                                          P2TNAME=$$ABERAJ
          File Protected:
            Select 0P17 (File Protect) message    ------>RETRY
                                                   RUN
                                          P2TNAME=$$ABERAJ
```

Figure 5 (Part 1 of 3). 8809-type Tape ERP A-Transient Operation

```
$$ABERAJ (continued):
        If error condition is Unit Check (continued)...
          Act according to error condition (continued):
          PE ID Burst Check:
            Select 0P44 (ID Burst Check) message  ------>RETRY
                                                   RUN
                                               P2TNAME=$$ABERAJ
          Environmental Data counter overflow:    ------>$$ABERAK
                                                  RFEXIT=RETRY
          Temporary Equipment Check:
            If device type is 9346                ------>RETRY
                                                   USER
            Else                                 ------>$$ABERAL
          Manual Unload:
            Select 0P43 (Tape Volume Change) msg
            RECORD_AND_MESSAGE (see below)
          Unsolicited Device End - Unit Check interrupt:
            Select 0P43 (Tape Volume Change) msg
            RECORD_AND_MESSAGE (see below)
          Otherwise
            Select 0P19 (Undetermined Erro) msg   ------>$$ABERRL
        If error condition is not Unit Check
          If error is Not-Ready-to-Ready Interrupt
            Select 0P43 (Tape Volume Change) msg  ------>$$ABERRL
          If error is Not Operational
            Select 0P31 (Not Operational) message ------>$$ABERRL
          If error is Channel Program Check
            Select 0P24 (Program Check) message   ------>$$ABERRL
          If error is Channel Protection Check
            Select 0P25 (Protection Check) message------>$$ABERRL
          Otherwise
            Select 0P19 (Undetermined Error) msg  ------>$$ABERRL


RECORD_AND_MESSAGE:
        If device type is 9346                    ------>$$ABERRL
        Else                                     ------>$$ABERAL
                                                RFEXIT=$$ABERRL
```

Figure 5 (Part 2 of 3). 8809-type Tape ERP A-Transient Operation

```
$$ABERAK:
        If processing Recording Request
          If recording has been done or device type is
             9346 (Note 1)
            Re-initialize PUB2 fields
            Post ECB for Recording Request        ------>IGNORE
          Else
            Select Record Type 90
        Else (processing an error condition)
          Select Record Type 91
        Build MDR body, except for sense/log data part
        Set fields in Recorder File Table to control
            recording and contents of MDR header
        If processing Recording Request
          Do SYSIO to get buffered log data and put
             it in the MDR body.                  ------>$$ABERA1
                                                RFEXIT=$$ABERAK
        Else
          Put sense data from ERBLOC in MDR body  ------>$$ABERA1



$$ABERAL:
        Build OBR                               ------>$$ABERA6
```

Figure 5 (Part 3 of 3). 8809-type Tape ERP A-Transient Operation

Notes:

1. $$ABERAK runs twice in the processing of a Recording Request. The first time, it transfers to $$ABERA1 to have an MDR recorded with the RFEXIT field in the Recorder File Table indicating itself ($$ABERAK). It sets a flag in the ERBLOC at that indicates it has completed its first run. When the recording A-transients transfer back to $$ABERAK (due to the RFEXIT value), $$ABERAK examines its flag and determines that it is in its second run. At that time, recording is finished and $$ABERAK finishes up the Recording Request processing, including posting the ECB on which the requester may be waiting.

3424, 3480, 3490, 3490, 9348 Tape Device ERP/Recording Request - $$ABERAY

Recording requests for 3480-Type Tape Devices are not honored, since all recording is done by tape device ERP in response to unit checks. Therefore, $$ABERAY simply re-initializes PUB2 fields, posts the ECB for the Recording Request and returns to the supervisor.

Device ERP is completely done in the supervisor and is discussed in *VSE/Central Functions Diagnosis Reference: Supervisor.*

2501, 2520, 2540 Card Device and 3881 OMR Recovery - $$ABERRX, $$ABERRY

$$ABERRY receives control from $$ABERRA and tests for the following conditions:

CSW bit 44 - Channel Data Check

- Action: If initial selection, one retry. If error persists, exits to the message writer with the allow retry bit on. If read data transfer, exits to the message writer for operator intervention. If punch data transfer, one retry. If error persists, exits to the message writer with the allow retry bit on.
- Message: 0P28 CHAN DTCHK.

CSW bit 47 - Channel Chaining Check (2540, 2520P and 3881 only)

- Action: Exits to the message writer with the allow retry bit on.
- Message: 0P19 UNDET ERR.

Sense byte 0, bits 6 and 7 (2501): Bits 5 and 7 (2540, 3881); bits 4, 5, and 6 (2520, on write command), Bit 6 (2520, not on write command).

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 3 - Equipment Check.

- Action: Reader - Exits to the message writer for operator intervention. Punch - If initial selection, continues testing sense bits. Otherwise, if return equipment check bit in CCB is on, turns on the passback bit, posts CCB (equipment check) and exits to message writer. If return equipment check bit is not on, exits to the message writer with the allow ignore bit on. For 2520, Byte 0, bit 7 indicates punch check.
- Message: 0P10 EQUIP CHK.

sense byte 0, bit 1 - Intervention Required.

- Action: If device end without channel end, ignores this condition. Otherwise exits to the message writer for operator intervention.
- Message: 0P08 INTERV REQ.

sense byte 0, bit 2 - Busout Check.

- Action: If initial selection, one retry. If error persists, exits to the message writer with the allow retry bit on. If the device is a 2540P, also one retry even when not initial selection.
- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 4 - Data Check (cannot occur on 2520 punch or 3881).

- Action: Exits to the message writer for operator intervention.
- Message: 0P11 DATA CHECK.

sense byte 0, bit 5 - Overrun (cannot occur on 2540 or 2520 punch or 3881).

- Action: Exits to the message writer for operator intervention.
- Message: 0P14 OVERRUN.

sense byte 0, bit 0 - Command Reject.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P18 COMM REJCT.

sense byte 0, bit 6 - Unusual Command Sequence (2540R or 3881).

- Action: Posts CCB and continues, ignoring the condition.
- Message: 0P18 COMM REJCT.

sense byte 0 = 00.

- Action: Exits to the message writer with the allow retry bit on.
- Message: 0P19 UNDET ERR.

CSW bit 47 - Channel Chaining Check (not 2540P, 2520P, or 3881).

- Action: Exits to the message writer for operator intervention.
- Message: 0P14 OVERRUN.

2560 Card Device Recovery - $$ABERRX, $$ABERRY

$$ABERRY receives control from $$ABERRA and tests for the following conditions:

CSW bit 44 - Channel Data Check

- Action: Exits to the message writer and then the task is canceled. If initial selection, one retry first.
- Message: 0P28 CHAN DTCHK.

CSW bit 47 - Channel Chaining Check.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 2.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 3 - Equipment Check.

- Action: Posts CCB (byte 3, bit 3). Checks CCB whether user wants return. If this is the case, exits to the message writer with the passback bit on. If the user did not specify return, exits to the message writer for operator intervention (the OPINT bit in the ERBLOC is on).
- Message: 0P10 EQUIP CHK.

sense byte 0, bit 1 - Intervention Required.

- Action: Exits to the message writer for operator intervention.
- Message: 0P08 INTERV REQ.

sense byte 0, bit 4 - Data Check.

- Action: Posts CCB (byte 3, bit 3). Checks in the CCB whether the user wants return. If this is the case, exits to the message writer with the passback bit on. If the user did not specify return, exits to the message writer for operator intervention (OPINT bit is on).
- Message: 0P11 DATA CHECK.

sense byte 0, bit 0 - Command Reject.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P18 COMM REJCT.

sense byte 0, bit 6 - No Card Available.

- Action: Posts CCB and exits to the message writer with the allow ignore bit on.
- Message: 0P18 COMM REJCT.
  0P08 INTERV. REQ. (For punch operations only).

sense byte 0 = 00.

- Action: Exits to the message writer with the allow retry bit on.
- Message: 0P19 UNDET ERR.

If the device is not operational, control is passed to $$ABERRA, with the repositioning required bit reset. Eventually the task is canceled.

PRT1 Printer Recovery - $$ABERRF, $$ABERRP, $$ABERRQ

PRT1 printers are 3211, 3203-4/5, 3262-1, 3263, 3289-4, and 4245.

Control is normally passed from $$ABERRA, but when counter overflow or CE mode recording has taken place, the recording A-transients $$ABERA1 and $$ABERA2 may pass control to $$ABERRF. It tests for the following conditions:

sense byte 0, bit 7 - Channel 9 Overflow

- Action: Posts CCB/IORB and continues error testing.

  There is also a storage resident test for channel 9 overflow (with no other bits posted in sense byte 0).

CSW bit 44 - Channel Data Check.

- Action: If initial selection, one retry. If error persists, exits to the message writer and then the task is canceled. If channel end, exits to the message writer with the allow ignore bit on.
- Message: 0P28 CHAN DTCHK.

sense byte 0, bit 3 - Equipment Check.

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 0 = Command retry
  6 = Mechanical motion or controller check
  1 = Print check

2 = Print quality
3 = Line position.

If none of these bits are on, a transparent synchronization check or a train overload condition occurred. Exits to resident routines (via recording A-transients) to continue processing.

- Message: 0P10 EQUIP CHK.

sense byte 0, bit 1 - Intervention Required.

- Action: Exits to the message writer for operator intervention (Intervention required bit on). The operator must make the device ready to continue processing.

  A forms check or an interlock condition has occurred on a PRT1. The probable causes are:

  Carriage detent switch off
  Paper jammed or forms torn
  Out of paper
  Stacker full
  Gate not latched
  Train not positioned
  Stop key active
  Vacuum low
  Train overload.

- Message: 0P08 INTERV REQ.

sense byte 0, bit 2 - Busout Check.

- Action: If not with FCB parity check (sense byte 3, bit 2), one retry. If error persists, exits to the message writer and then the task is canceled.

  If with FCB parity check, exits to the message writer with the allow retry bit on.

- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 4 - Data Check.

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 1 = Print check
  3 = Line position

  If neither of these bits are on, exits to the message writer with the allow ignore bit on.

- Message: 0P11 DATA CHEK.

sense byte 0, bit 5 - Buffer Parity Check.

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 0 = Command retry
  3 = Line position

  If neither of these bits are on, and if the error occurred on a write command, posts it in the associated CCB.

If return is specified in the CCB (byte 2, bits 5 and 6 on), turns on the passback bit and exits to the message writer and then control is returned to user.

If return is not specified, exits to the message writer with the allow ignore bit on, to provide a message that allows the operator to respond with IGNORE or CANCEL.

If the error occurred on a buffer read command, returns (via recording A-transients) to resident routines to continue processing.

This A-transient also sets the RFEXIT field of the recorder file table, which will be used later to determine the exit from this or following A-transients.

- Message: 0P33 UCB PARITY.

sense byte 0, bit 0 - Command Reject.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P18 COMM REJCT.

sense byte 0, bit 6 - Load Check.

- Action: An error condition occurred when either the UCSB or the FCB was loaded. Exits to the message writer and then the task is canceled.
- Message: 0P41 LOAD CHECK.

sense byte 0 = 00

- Action: Exits to the message writer with the allow retry bit on.
- Message: 0P19 UNDET ERR.

sense byte 1, bit 0 - Command Retry.

- Action: For message 0P10, a print line buffer parity check occurred. The recovery procedure depends on the users error option for command retry (CCB/IORB byte 2, bit 5).

  1. If the command retry option is specified, reissues the failing CCW. If the retry is unsuccessful, the error is considered irrecoverable. Issues a skip to channel 0 (hex 83) command to allow the carriage to perform the suppressed carriage motion command. Then exits to the message writer and the task is canceled.
  2. If the command retry option is not specified, issues a skip to channel 0 (hex 83) command to allow the carriage to perform the suppressed carriage motion command. Then exits to the message writer and the task is canceled.

  For message 0P33, a UCSB parity check occurred. If return is specified in the CCB (byte 2, bit 6 on), turns on the passback bit and exits to the message writer and then control is returned to user. If return is not specified, issues a skip to channel 0 (hex 83) command to allow the carriage to perform the suppressed carriage motion command, then exits to the message writer and the task is canceled.

  Note: When control is returned to the user, the action depends on the users error routine to reload the UCS buffer. After the UCS buffer is reloaded, the failing CCW can be reissued.

sense byte 1, bit 1 - Print Check.

- Action: For message 0P10, a print check occurred for one or more of the following reasons:

  Hammer fire checks
  Synchronization check
  Coil protection.

  For message 0P11, a print check occurred because of an unprintable character. In both cases (0P10 and 0P11), the following action is taken:

  Posts the error in the CCB. If return is specified in the CCB (byte 2, bit 6 on), turns on the passback bit and exits to the message writer and then control is returned to user. If return is not specified, exits to the message writer with the allow ignore bit on.

sense byte 1, bit 2 - Print Quality

- Action: A print quality check occurred because the platen failed to advance or retract, or because excessive ribbon motion or ribbon skew or both were detected. Posts the error in the CCB. If return is specified in the CCB (byte 2, bit 6 on), turns on the passback bit and exits to the message writer and then control is returned to user. If return is not specified, then exits to the message writer with the allow ignore bit on.

sense byte 1, bit 3 - Line Position.

- Action: For message 0P10, the probable causes are:

  Carriage failed to move
  Carriage sequence check
  Carriage stop check.

  For message 0P11, the probable cause is: A SKIP command to a channel code that was not loaded into the FCB.

  For message 0P33, the probable cause is: Data transmision error in loading the FCB (detected by a parity check).

  In each case, posts the error in the CCB. If return is specified in the CCB (byte 2, bit 6 on), turns on the passback bit and exits to message writer. Then control is returned to the user.

  If return is not specified, exits to the message writer with the allow ignore bit on (0P10 and 0P11). For message 0P33, the task is canceled eventually.

sense byte 1, bit 6 - Mechanical Motion or Controller Check.

- Action: Exits to the message writer and then the task is canceled.

4248 (native mode) Printer Recovery - $$ABERBF

$$ABERBF performs SDR counter update and error recovery procedures for printers which make use of an action code, in sense byte 1, to initiate the error recovery. $$ABERBF gets control from $$ABERRJ.

Recovery actions on action code in sense byte 1.

Action code X'00'

- Represents a reset condition. No special action is required. When sense byte 0 is not zero, MSG 0P19 UNDET ERR is given and the job is cancelled.

Action code X'01'

- Means error detected in the data sent to the printer (command reject, data check, load check). No printing or mechanical motion has occured.
- Actions:
  - Issue MSG, depending on SNS byte 0, to allow retry.
  - If not first time, log OBR record.
- Messages:
  - 0P10 EQUIP CHK equipment check
  - 0P09 BUSOUT CHK bus out check
  - 0P18 COMM REJCT command reject
  - 0P11 DATA CHECK data check
  - 0P41 LOAD CHECK load check

Action code X'02'

- Means intervention required condition, caused by an abnormal error such as: Forms jam, Print check.
- Actions:
  - The error is logged.
  - Issue MSG.
  - The failing CCW is retried.
- Messages:
  - 0P07A OPER VERFY verify the printing

Action code X'03'

- Means normal intervention required condition (end of forms, stop key, stacker full).
- Actions:
  - Issue MSG: 0P08A INTERV REQ intervention required.

Action code X'04'

- Means that an error was detected in the channel attachment hardware or in the micro processor.
- Actions:
  - The error is logged.
  - The operation is retried one time.
  - If error persists, give MSG and cancel.
- Messages:

- 0P63 UNRECV ERR unrecoverable error

Action code X'05'

- Means that an error was detected in the error handling functions within the subsystem.
- Actions:
  - The error is logged.
  - Issue MSG.
  - Restart from next CCW.
- Messages:
  - 0P64 MAINT REQD maintenance required

Action code X'06'

- Means micro code failure.
- Actions:
  - The error is logged.
  - Issue MSG.
  - Restart from next CCW.
- Messages:
  - 0P63 UNRECV ERR unrecoverable error

Action code X'07'

- Means micro code hang condition.
- Actions:
  - The error is logged.
  - Issue MSG.
  - Restart from next CCW.
- Messages:
  - 0P06A IML REQD IML is required

Action code X'0A'

- Means channel 9 was detected on the last operation. This action code is described here just for completeness. $$ABERBF is not entered, the function will have already been performed by the supervisor.
- Actions:
  - Post channel 9 in the CCB.
  - Go to exit ignore.

Action code X'0B'

- Means probable performance degradation due to maintenance problem.
- Actions:
  - The error is logged.

  - Issue MSG.
  - Restart from next CCW.
- Messages:
  - 0P64 MAINT REQD maintenance required

Action codes not described above will lead to cancel with MSG: 0P19 UNDET ERR.

other messages issued:

- 0P19 UNDET ERR undetremined error
- 0P20 RECOVY ERR error on recovery
- 0P24 PROG CHECK channel program check
- 0P25 PROT CHECK channel protection check
- 0P28 CHAN DTCHK channel data check
- 0P31 DVC NOT OP device not operational

3277, 3278, 3279 Terminal Recovery - $$ABERP1, $$ABERP5

These phases belong to the separate program product BTAM.

3505, 3525 Card Device Recovery - $$ABERRG

$$ABERRG receives control from $$ABERRA and tests for the following conditions:

CSW bit 45 and 46 - Channel or Interface Control Check.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

CSW bit 44 - Channel Data Check.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P28 CHAN DTCHK.

CSW bits 32, 33, or 34 on or unit check bit in CSW status byte off or sense byte 0, bit 5 on

- Action: If any of these occurs, there must be a hardware error. Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 7 - Permanent Error (Error Bypass Key).

- Action: Operator pressed the permanent error bypass key instead of having recovery performed. If return permanent error bit in CCB is on, turns the passback bit on, posts CCB, and exits to the message writer. Otherwise exits to the message writer and then the task is canceled.
- Message: 0P49 PERM ERROR.

sense byte 0, bit 3 - Equipment Check.

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 0 = Permanent error
  1 = Automatic retry
  3 = Retry after intervention complete

  If none of these bits are on, exits to the message writer and then the task is canceled.

- Message: 0P10 EQUIP CHK.

sense byte 0, bit 6 - Abnormal Format Reset.

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 0 = Permanent error
  1 = Automatic retry
  3 = Retry after intervention complete

  If none of these bits are on, exits to message writer and then the task is canceled.

- Message: OP48 FORMAT RST.

sense byte 0, bit 1 - Intervention Required

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 0 = Permanent error
  1 = Automatic retry
  3 = Retry after intervention complete

  If none of these bits are on, exits to the message writer and then the task is canceled.

- Message: 0P08 INTERV REQ.

sense byte 0, bit 2 - Busout Check.

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 0 = Permanent error
  1 = Automatic retry
  3 = Retry after intervention complete

  If none of these bits are on, exits to message writer and then the task is canceled.

- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 4 - Data Check.

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 0 = Permanent error
  1 = Automatic retry
  3 = Retry after intervention complete

If none of these bits are on, exits to message writer and then the task is canceled.

- Message: 0P11 DATA CHECK.

sense byte 0, bit 0 - Command Reject.

- Action: Checks the following bits of sense byte 1; if a bit is on when tested, the system takes the action indicated:

  bit 0 = Permanent error
  1 = Automatic retry
  3 = Retry after intervention complete

  If these three bits are off, exits to the message writer and then the task is canceled.

- Message: 0P18 COMM REJCT.

None of the above.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 1, bits 4, 5, 6, or 7

- Action: If any of these is on, exits to the message writer and then the task is canceled.
- Message: Depends on bit setting in byte 0.

sense byte 1, bit 0 - Permanent Error.

- Action: If Return Permanent Error bit in CCB is on, turns passback bit on, posts CCB and exits to the message writer. Otherwise exits to the message writer and then the task is canceled.
- Message: Depends on bit setting in byte 0.

sense byte 1, bit 3 - Retry after intervention complete.

- Action: Exits to the message writer for operator intervention. (Intervention required bit on).
- Message: Depends on bit setting in byte 0.

sense byte 1, bit 1 - Automatic Retry.

- Action: Retries failing CCW once. If successful, continues normal program execution. If unsuccessful, and if Return Permanent Error bit in CCB is on, turns the passback bit on, posts CCB, and exits to the message writer. Otherwise exits to the message writer and then the task is canceled.
- Message: Depends on bit setting in byte 0.

3540 Diskette Device Recovery - $$ABERR7

$$ABERR7 receives control from $$ABERRA and performs three functions:

1. Determines the proper error recovery action to be taken;
2. Determines proper message code for the message writer;
3. Updates the statistic counters in the PUB2 table entry for the device.

$$ABERR7 tests for the following conditions:

CSW bits 45 and 46 - Channel or Interface Control Check.

- Action: Exits to the message writer via recording phase, and then the task is canceled.
- Message: 0P19 UNDET ERR.

CSW bit 44 - Channel Data Check.

- Action: Exits to the message writer, then the task is canceled.
- Message: 0P28 CHAN DTCHK.

CSW bit 38 - Unit Check.

- Action: If this bit is 0, exits to the message writer then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 0, bits 5, 6, or 7

- Action: If any of these is on, exits to the message writer with retry and allow ignore bits off. Exit to message writer is via recording phase for recording of an I/O device error record.
- Message: 0P19 UNDET ERR.

sense byte 0, bit 3 - Equipment Check.

- Action: Determines further action by checking the bits in sense byte 1.
- Message: 0P10 EQUIP CHK.

sense byte 0, bit 1 - Intervention Required.

- Action: If motion malfunction, sets the bit to write an I/O device error record. Determines further action by checking the bits in sense byte 1.
- Message: 0P08 INTERV REQ.

sense byte 0, bit 2 - Busout Check.

- Action: Sets bit to write an I/O device error record then determines further action by checking the bits in sense byte 1.
- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 4 - Data Check.

- Action: Determines further action by checking the bits in sense byte 1.
- Message: 0P11 DATA CHECK.

sense byte 0, bit 0 - Command Reject.

- Action: Sets on the allow ignore bit and exits to the message writer.
- Message: 0P18 COMM REJCT.
- System Action: Checks bits in sense byte 1.

sense byte 1, bits 5, 6, or 7

- Action: If any of these is on, sets off retry and ignore bits, and exits to the message writer via recording phase for I/O device error recording.
- Message: 0P19 UNDET ERR.

sense byte 1, bit 0 - Permanent Error.

- Action: Posts irrecoverable I/O bit in CCB. If a Data Check, posts Data Check bit in CCB. If return to user is requested, exits to the message writer with allow ignore bit on, via I/O device error recording A-transient. If return is not requested, exits to the message writer with allow ignore bit and retry bit off, via I/O device error recording A-transient.

sense byte 1, bit 3 - Retry After Intervention.

- Action: If I/O device errror recording is wanted, exits to the message writer with action and retry bits on, via recording phase. Else exits to the message writer with action and retry bits on.

sense byte 1, bit 1 - Automatic Retry.

- Action: Retries the error 10 times, updating the statistics counter each time. If no recovery, then indicates permanent error (see Permanent Error Action). If statistics counter overflows, retries after recording phase for I/O device error.

sense byte 1, bit 4 - Special Record Transferred.

- Action: Sets on special record transferred bit in CCB. Exits to the message writer with ignore bit on.
- Message: 0P55 SPEC REC.

None of the above

- Action: Exits to the message writer with retry and ignore bits off, via recording phase.
- Message: 0P19 UNDET ERR.

3800 Printer Recovery - $$ABERBA, $$ABERBC, $$ABERBD

$$ABERBA receives control from $$ABERRA. If the CCB is not available or the address in the CSW is zero, the allow ignore bit will have been set by PIOCS; otherwise the allow retry bit will have been set. The A-transient tests the following bits:

CSW bit 44 - Channel Data Check.

- Action: If retried five times, the allow ignore bit is set; exits to $$ABERBD to write an I/O device error record and then exits to $$ABERRL to write the error message. Otherwise the statistics counter is updated by one. If the statistics counter overflows, exits to $$ABERBD to write an I/O device error record, and returns to the supervisor to allow retry.

  If initial selection or a read or sense command, exits the supervisor to allow retry.

  If it is a write command and not initial selection, exits to $$ABERRL to write an error message after turning off both the allow retry and allow ignore bits.

- Message: 0P28 CHAN DTCHK.

sense byte 0, bit 3 - Equipment Check.

- Action: Checks bit 2 of sense byte 1. If this is on, the internal error log is full and exit is to $$ABERBC to create an MDR record. Exit from there goes to $$ABERA1 to write the record. Return to the supervisor is from $$ABERA1, and no error message is written.

  If not error log full condition, the intervention required bit is set and exits are set to go to $$ABERBD, $$ABERA1, and $$ABERRL to create the I/O device error record, write the record on SYSREC, and to write the error message. Returns to the supervisor to retry the failing CCW unless system restart is required (sense byte 3, bit 4), or if the CCW is not available.

  If system restart is required or if the CCW is not available, the allow ignore bit is set on and the allow retry bit is set off. If system restart is required, the intervention required bit in the ERQFLG field of the ERBLOC is set off and intervention required is posted in the PUB.

  If system restart is not required and the CCW is available, the field in the CCB containing the CCW address in the CSW is updated if the command chain retry bit is on and the channel appendage or the OLTEP appendage are not on or appendage exit is not allowed.

- Message: 0P08 INTERV REQ.

For all equipment check errors, only intervention required messages are produced, and the operator must make the device ready before processing can continue.

sense byte 0, bit 1 - Intervention Required.

- Action: If sense byte 4 is a 32 (CFS misfold), 40 (burster trimmer jam), 41 (no burst check), or a 42 (burster stacker jam), the proper statistics counter is updated by one. If the statistics counter overflows (count of 15) or sense byte 4 has a value greater than hex 0F (and not 32, 40, 41, or 42), exit will be to $$ABERBD and $$ABERA1 to create and write an I/O device error record. All conditions will exit to $$ABERRL to write the intervention required message, and will have the intervention required bit in the ERBLOC set on.

If the system restart is required or if the CCW is not available, the allow ignore bit is set on, and the allow retry bit is set off. If system restart is required, the intervention required bit is set off in the ERBLOC and set on in PUBCSFLG.

If system restart is not required and the CCW is available, the field in the CCB containing the CCW address in the CSW is updated if the command chain retry bit is on and the channel appendage or OLTEP appendage bits are not on or appendage exit is not allowed.

- Message: 0P08 INTERV REQ.

sense byte 0, bit 2 - Busout Parity.

- Action: Will retry 5 times, and for each retry will exit to $$ABERBD to create a temporary busout parity I/O device error record, returning to the supervisor with retry bit on, and no message issued. If retried 5 times, will exit to $$ABERBD to create a permanent busout parity error record, and exit to $$ABERRL, the message writer, with the allow ignore bit on.
- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 0 - Command Reject.

- Action: Exits to the message writer and then task is canceled.
- Message: 0P18 COMM REJCT.

sense byte 0, bit 6 - Load Check.

- Action: An error condition occurred in loading. Sense bytes 1 and 2 further define the cause. Exits to the message writer and then the task is canceled.
- Message: 0P41 LOAD CHECK.

sense byte 0, bit 4 - Data Check.

- Action: An error condition occurred, with sense byte 1 further defining the cause. Exits to the message writer with the allow ignore bit on.
- Message: 0P11 DATA CHECK.

None of the above.

- Action: Exits to the message writer with allow retry bit on.
- Message: 0P19 UNDET ERR.

3886 OCR (Optical Character Reader) Recovery - $$ABERAN

$$ABERAN receives control from $$ABERRA. It determines the error recovery action required and updates error statistics in the PUB2 table for the device. If recording on IJSYSRC is required (OBR sense bit on, or counter overflow) passes control to recording phases. The first recording phase fetched is $$ABERA4. If no recording is necessary, exits directly to the supervisor or the message writer, depending on the error condition. $$ABERAN tests for the following conditions:

CSW bits 45 and 46 - Channel or Interface Control Check.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

CSW bit 44 - Channel Data Check.

- Action: One retry. If error persists, exits to the message writer and then the task is canceled.
- Message: 0P28 CHAN DTCHK.

CSW bits 32, 33, or 34

- Action: If any of these is on, exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

CSW bit 38 - Unit Check.

- Action: If this bit is 0, exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

Sense Byte 0, Bits 4 or 5 or Byte 1, Bits 0, 3, or 5

- Action: If any of these bits is on, posts permanent error in the CCB. Exits to the message writer with the allow ignore bit on. Exits via recording phases for recording of a unit check condition record.
- Message: 0P19 UNDET ERR.

Sense byte 0, bit 7 - Recognition Control Program Error.

- Action: Posts permanent error in the CCB. Exits to the message writer with the allow ignore bit on. Exit is taken to the message writer via recording phases for recording of a Unit Check Condition record.
- Message: 0P53 RCP ERROR.

sense byte 0, bit 6 - Non Initialized.

- Action: Posts permanent error in the CCB. Exits to the message writer with the allow ignore bit on.
- Message: 0P54 NOT ICPL'D

sense byte 0, bit 3 - Equipment Check.

- Action: If mark check is not on, and the error occurred on a load format or scan command, one retry. Before returning to the supervisor for the retry, checks whether recording is required (OBR sense bit, byte 1, bit 7 on). If it is, exits via recording phases. Otherwise updates error statistics before return to the supervisor (via recording phases if counter overflow occurred).

  If the error persists or if retry is not permitted, posts permanent error in the CCB and exits to the message writer with the allow ignore bit on. Mark check is also posted in the CCB. If the CBR sense bit is on, exits via recording phases. If this bit is not on, updates error statistics and exits to the message writer (via recording phases if counter overflow occurred).
- Message: 0P10 EQUIP CHK.

sense byte 1, bit 6 - Non Recovery Error.

- Action: Posts non-recovery error in the CCB, updates a statistics counter, and exits to the message writer with OPINT and passback bits on. If recording is required because of counter overflow, exits via recording phases.
- Message: 0P35 NON RECOV.

sense byte 1, bit 4 - Incomplete Scan.

- Action: One retry. Before returning to the supervisor for the retry, updates error statistics. If recording is required because of counter overflow, exits via recording phase. If the error persists, posts incomplete scan in the CCB and takes supervisor exit for ignore. Updates error statistics and exits. If recording is required because of counter overflow, exits via recording phases.
- Message: none

sense byte 1, bit 1 - Mark Check Error.

- Action: Posts mark check error in the CCB, updates error statistics, and exits (via recording phases if counter overflow occurred) to the message writer with OPINT and passback bits on.
- Message: 0P51 MARK CHECK.

sense byte 0, bit 1 - Intervention Required.

- Action: Updates error sttistics. Exits to the message writer with operator action required and allow retry (if not ending status, or load format command) or passback (otherwise) bits on. If counter overflow occurred, exits via recording phases for recording.
- Message: 0P08 INTERV REQ.

This condition can occur as ending status only when an eject command has been issued. In this case, the command is not to be reissued.

sense byte 0, bit 2 - Busout Check.

- Action: One retry. Exits to supervisor for retry via recording phases to have the error recorded. If error persists, posts a permanent error in the CCB and exits to message writer with the allow ignore bit on. Exits to the message writer via recording phases to have the error recorded.
- Message: 0P09 BUSOUT CHK.

sense byte 0, bit 0 - Command Reject.

- Action: Posts permanent error in the CCB. Exits to the message writer with the allow ignore bit on.

- Message: 0P18 COMM REJCT.

sense byte 1, bit 2 - Invalid Format.

- Action: Posts permanent error in the CCB. Exits to the message writer with the allow ignore bit on.
- Message: 0P52 INV FORMAT.

None of the above.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

5424, 5425 Card Device Recovery - $$ABERRX, $$ABERRY

$$ABERRY receives control from $$ABERRA. $$ABERRY tests for the follwing conditions:

CSW bit 44 - Channel Data Check.

- Action: Exits to the message writer and then the task is canceled. If initial selection, one retry first is attempted.
- Message: 0P28 CHAN DTCHK.

CSW bit 47 - Channel Chaining Check.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

sense byte 0, bits 2, 4, 5, or 7.

- Action: If any of these is on, exits to the message writer and then the task is canceled.
- Message: 0P19 UNDET ERR.

  This message is also issued if the equipment check bit is on (see below) and sense byte 1 is zero.

sense byte 0, bit 3 - Equipment Check, and

sense byte 1, bit 7 - Punch Data Check (5424 only).

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P58 PCH DTCHK.

sense byte 0, bit 3 - Equipment Check, and

sense byte 1, bit 2 - Cycle Steal Overrun (5424 only).

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P10 EQUIP CHK.

sense byte 0, bit 3 - Equipment Check, and

sense byte 1, any of bits 0, 1, 3, 4.

- Action: Posts CCB (byte 3, bit 3). Checks CCB whether user wants return. If this is the case, exits to the message writer with the passback bit on. If the user did not specify return, exits to the message writer for operator intervention (OPINT bit is on).
- Message: 0P10 EQUIP CHK.

sense byte 0, bit 1 - Intervention Required, or

sense byte 0, bit 3, and

sense byte 1, bit 6.

- Action: Exits to the message writer for operator intervention.
- Message: 0P08 INTERV REQ.

sense byte 0, bit 0 - Command Reject.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P18 COMM REJCT.

sense byte 0, bit 6 - No Card Available.

- Action: Exits to the message writer and then the task is canceled.
- Message: 0P18 COMM REJCT.

If the device is not operational, control is passed to $$ABERRA, with the repositioning required bit reset. Eventually, the task is canceled.

Recording - $$ABERA1, $$ABERA2, $$ABERA6

These A-transients take as input an SDR which is mostly built and and write it onto IJSYSRC if possible.

$$ABERA6 does one of the following, depending on values in the ERBLOC:

- Add the OBR device type code (OS device class/type) to an OBR.
- Add the subid field information to an MDR.
- Enqueue the recorder file.
- Dequeue the recorder file.

In the first two cases, $$ABERA6 exits as follows:

- If ERP is processing an error condition, $$ABERA6 transfers to $$ABERA1.
- If ERP is processing a Recording Request, $$ABERA6 transfers to $$ABERA3 if IJSYSRC is on a CKD device and to $$ABERA9 if IJSYSRC is on an FBA device.

In the case of the enqueue/dequeue calls of $$ABERA6, $$ABERA6 returns to the supervisor, requesting IGNORE processing.

$$ABERA1 builds the standard 24-byte header and does the following:

- If IJSYSRC is on a CKD device, it reads the recorder file header record and determines whether the record passed to it will fit in the file. If so, the record is appended to the standard 24-byte header. The record is then written onto the next available space in the file, and the recorder file record is updated and rewritten onto the file. SVC76 is used to write the record.
- If IJSYSRC is on an FBA device, it reads the recorder file header record and determines whether the record passed to it will fit in the file. If so, the record is appended to the standard 24-byte header and put in the next available space in the output buffer. The block is written onto the file each time a record is added. Whenever a new block is allocated, the recorder file header record is updated and rewritten onto the file. SVC76 is used to write the block.

$$ABERA1 passes control to the A-transient specified in the RFEXIT field. For the following conditions, control is passed to $$ABERA2 to write a message on SYSLOG:

- I/O-error while accessing IJSYSRC.
- IJSYSRC is full.
- Last track of IJSYSRC is being used (CKD) or less than 16 blocks available on IJSYSRC (FBA).

If IJSYSRC is not ready or is being accessed by EREP, exit is taken without recording.

$$ABERA2 causes one of the following to be written on SYSLOG:

```
0T00I THRESHOLD ON RECORDER FILE REACHED
0T03I ERROR ON RECORDER FILE AT cchhr/blockno
0T05E RECORDER FILE FULL  --  RUN EREP
```

Unit Check Recording for Non-Disk, Non-Tape Devices

This section discusses $$ABERRJ, $$ABERJ1, and $$ABERRA.

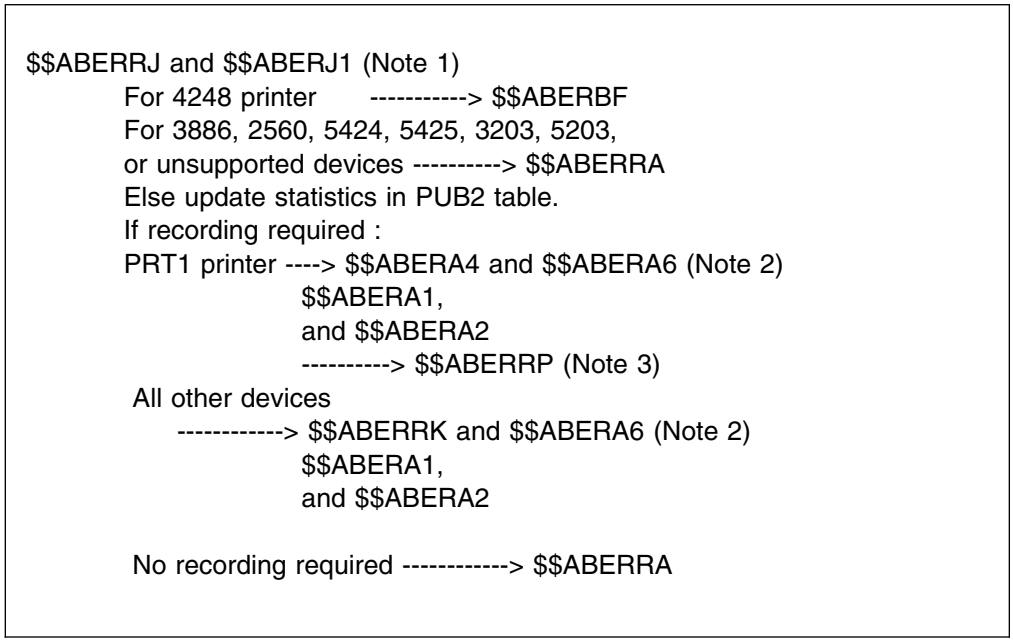

Figure 6 (Part 1 of 2). Unit Check Recording for Miscellaneous Devices

```
$$ABERRA   If error on recovery, device not operational,
           protection or program check
               ------------> $$ABERRK and $$ABERA6 (Note 2)
                             $$ABERA1,
                             and $$ABERA2
           else analyze device type and exit for
           PRT1            ----------->  $$ABERRF (Note 4)
           3800            ----------->  $$ABERBA (Note 5)
           Console Printer Keyboard
           and Display Console ------>  $$ABERRW
           3505,3525       ----------->  $$ABERRG
           1403,1443,3203,5203,1424,2596
           2501,2520,2540,3881,2560,5424
           and 5425        ----------->  $$ABERRY,$$ABERRX
           MICR            ----------->  $$ABERRS
           1287/1288 OCR   ----------->  $$ABERRT
           TP Device (BTAM-ES)------->  $$ABERP1,$$ABERP5
           2245            ----------->  $$ABERRH
           3886            ----------->  $$ABERAN
           3895            ----------->  $$ABERAO,$$ABERAP,
                                         $$ABERAQ,$$ABERAR
           3540            ----------->  $$ABERR7

           The A-transients do the necessary recovery action and
           set up retry, ignore or cancel in the RFEXIT field.
           If recording is required control is given to the
           record builder phases for
           PRT1            ----------->  $$ABERA4,$$ABERRP,
                                         $$ABERRQ
           3800            ----------->  $$ABERBD,$$ABERBC
           3895            ----------->  $$ABERAQ,$$ABERAR
           others          ----------->  $$ABERRK
           The record builder invoke for recording
                        -----------> $$ABERA1,
                                     and $$ABERA2

           If a message is required
                      ----------> Message Writer
                      ----------> ERPEXIT
```

Figure 6 (Part 2 of 2). Unit Check Recording for Miscellaneous Devices

Notes:

1. $$ABERRJ updates the statistics in the PUB2 table, except for the 3800, 4248 printers and the 3886 OCR. In the latter case, control goes to $$ABERRA without updating the counters as this function is performed by the 3886 error recovery, $$ABERAN. The 3800 printer control goes to $$ABERBA to update the statistics in the PUB2 table and for error analysis. The 4248 printer control goes to $$ABERBF to update the statistics in the PUB2 table and for error analysis.

   After the updating, a test is made to determine whether recording on IJSYSRC is required. Otherwise control goes to $$ABERRA. If recording is necessary,

control is passed to $$ABERA4 for a PRT1 printer or to $$ABERRK for other devices. This is the case when the following conditions exist:

- The device is in diagnostic mode and the limit count is not yet reached.
- The device is in intensive mode, the error is initial, and limit count not reached.
- A counter in the PUB2 table is full.

In the first two cases, bit hex 40 in RFFLAGS3 in the recorder file table is set and the limit count is reduced by one. In the third case, bit RECDSG (hex 20) is set.

If it is found that the device is in the diagnostic or intensive mode and the record limit has been reached (equal to zero), the device is reset to normal mode.

2. Except for the conditions specified below, a device error record is built by $$ABERA4 or by $$ABERRK. The statistics in the PUB2 table are reset. $$ABERA6 adds the OS/VS device class and type code to the record.

   For the following message codes, no record is build and control is given to $$ABERRP when the device is a PRT1 printer or to the message writer $$ABERRL :

   hex 08 Intervention required
   hex 17 File protect violation
   hex 18 Command reject
   hex 24 Program check
   hex 25 Protection check
   hex 26 Invalid seek address
   hex 29 Backspace to load point
   hex 31 Device not operational
   hex 32 Non-compatible tape on drive
   hex 34 Batch numbering bit off
   hex 58 Punch data check

3. By inspecting the sense bytes, $$ABERRP first determines which type of PRT1 MDRs are to be built by this phase and $$ABERRQ. It uses bits 1, 2, and 3 of RFFLAGS4 in the recorder file table to register which records are to be prepared.

   A forms control buffer off-load record (RFFCB on) is required for the following error conditions:

   - FCB parity check
   - Load check on FCB load command
   - Data check with line position.

   A print line buffer or check read buffer off-load record (RFPLB on) is required for the following error conditions:

   - Mechanical motion/controller check
   - Data check without line position
   - Data check with line position and print check
   - PLB parity check
   - Coil protection check
   - Hammer fire check.

A UCS buffer off-load record (RFUCB on) is required for the following error conditions:

- Load check on UCSB load command
- Mechanical motion/controller check
- Data check without line position.
- Data check with line position and print check
- UCB parity check.

An any hammer on (AHO) off-load record is required for the following combination of error conditions (3289-4 printer only):

- Equipment check: byte 0 bit 3,
- Print check: byte 1 bit 1,
- Hammer fire check: byte 3 bit 4, and
- Any hammer on check: byte 9 bit 1.

After the sense bytes have been inspected and the RFFLAGS4 flag byte in the recorder file table has been initialized, this flag byte is used to determine which MDR records are to be built and written. The PLB check read buffer off-load, the FCB off-load, and the AHO off-load records are built by $$ABERRP. $$ABERRQ builds the UCSB off-load records. Every time a record is completed, $$ABERA1 is called to append it to the standard 24-byte header and to write it on IJSYSRC.

To build the AHO record a check read command is issued on a 3289-4 to get a bit string that is 17 bytes long. Starting on the left, each bit of the string represents one of the 132 print positions. The last 4 bits of the 17th byte are not used and always zero. The significant 132 bits are zero except when sense byte 9 bit 0 indicates any hammer on' check. In that case the failing print position is marked by setting the associated bit in the bit string to 1.

To build the PLB record (not applicable to 3289-4, 3262, or 3263), a check read command is issued to read check information stored in each addressable position of the PLB. Then a read PLB command is issued to read in data from the PLB. The check information is scanned for error conditions. A maximum of 10 characters, on which a print error check occurred, are saved. The check information and the first ten PLB error positions are contained in the PLB record.

To build the FCB records, $$ABERRP first issues a Diagnostic Gate command, followed by a check read command, to obtain the eight bits of the Forms Control Address Register (FCAR). The FCB record is built (in case of 3211 only if the FCAR is at position 1) and written in two parts. They contain the FCAR and the carriage codes obtained from the FCB by issuing the read FCB command.

If for a 3211 the FCAR is not at position 1, the read FCB command is not issued, to avoid misalignment of the forms. Only one FCB record is built and written. This record does not contain the carriage codes. $$ABERRQ builds (and $$ABERA1 writes) the UCSB record in three parts. They contain the contents of the UCS Buffer, obtained via the READ UCSB command. See also items 4 and 5. The message 4E10I cannot be printed at this stage.

4. The PRT1 error recovery A-transient $$ABERRF uses bit 4 of RFFLAGS4 of the recorder file table to indicate that the error is to be ignored. When on, control goes to resident routines. When off, control is passed to the recovery message writer.

5. The 3800 recovery A-transient $$ABERBA analyzes the error, updates the statistics and causes transfer to $$ABERBD (outboard record builder) for statistics counter overflow, some intervention required conditions, equipment check errors, busout parity, and permanent channel data check errors. Control is transfered to $$ABERBC (MDR record builder) if an equipment check/error log-full condition is encountered. Exits from the 3800 error recovery A-transients are to the ERP message writer or to the ERPEXIT in supervisor.

R-Transients

R-transients perform the recording and preliminary recovery functions of machine checks, channel control and interface control checks (stated as channel checks in the following description).

The names of the R-transients are "$$RASTxx" and the phases are created by source files named "ILVRASxx".

Phase-Function Overview

Phase	Function
$$RAST00	Root phase for machine check, channel check and channel report word (CRW) handling
$$RAST01	Machine and channel check record builder
$$RAST02	Channel check handler root phase
$$RAST03	Machine check analysis and message scheduling
$$RAST04	Channel report word (CRW) handling
$$RAST05	Channel check recovery for 1050,1403, 2501, 3210, 3800, 1442, 2520, 3215, 3881, 1443, 2540,
$$RAST06	Channel check recovery for 3504,3505,3525,3540, and 3886
$$RAST07	Channel check recovery for 24xx tapes
$$RAST08	Machine and channel check record writer (CKD)
$$RAST10	Message writer for channel check recovery and CRW handling
$$RAST11	Message writer for machine checks
$$RAST12	Channel check recovery for 3420, etc., 8809, etc. tapes
$$RAST13	Message writer for storage error messages
$$RAST14	Channel check recovery for 3277, 3278, 3279 terminals
$$RAST15	Channel check recovery for 3705, 3791 terminals
$$RAST16	Machine and channel check record writer (FBA)
$$RAST18	(documented in the 3890 support manual)
$$RAST19	Channel check recovery for 3424, 3480
$$RASTXA	Channel check recovery for Page-Printers. Phase belongs to the separate program product VSE/PSF.

The RAS task consists of the supervisor resident RAS monitor and the RAS transients, which are loaded on request in the RAS transient area (RTA). The RTA is located after the resident machine check, channel check and CRW handler within the supervisor and has a length of 1200 bytes. The RAS monitor provides the following services needed by the RAS transients:

- Moves the error entry to the work ERPIB
- Fetches R-Transients into the RTA
- Schedules I/O requests from the RTA
- Dequeues I/O requests in error
- Dequeues defective page frames
- Frees I/O extended logout areas
- Provides an exit interface from R-Transients

An R-transient requests one of these services from the supervisor by setting fields in the RAS Monitor Table (RASTAB) and a return address in a register and branching to a linkage point in the supervisor. The address of the linkage point is in the SUPLINK field of RASTAB.

Resident RAS, RAS monitor and the RAS transients use for communication:

1. For machine checks
   - The machine check interruption code (MCIC) and the limited machine check logout area in low storage

   Both are queued by resident RAS into a machine check queue (which has 5 queue elements that are used in a wraparound mode) and thus made available for the RAS monitor and the RAS transients.
2. For channel checks
   - The work ERPIB (Error Recovery Procedure Information Block) copied from the RAS error chain within the ERBLOC area of the supervisor
   - The processor dependent I/O extended logout area pointed to by the ERPIB
3. For Channel Report Words
   - The Channel Report Word itself
   - The associated device (if available)
   - The TOD clock

   are queued into a CRW queue and thus made available for the RAS monitor and the RAS transients. The CRW queue has 5 elements that are used in a wraparound mode. Each queue element may hold up to 32 CRWs.
4. For recording and recovery handling
   - RAS linkage area (RASLINK)
   - RAS monitor table (RASTAB)
   - RAS record area (RTACOM) following the RTA

Machine Check Handling

When a machine check interrupt (MCI) occurs, hardware first logs the error in the machine check logout area in low storage (hex E8) and then retries the failure by CPU retry and ECC (error checking and correction). If the retry is successful, a recoverable MCI is recorded.

A hard MCI occurs when:

- CPU retry fails.
- No retry of interrupted instruction takes place.
- Storage failure is permanent.

In the event of a hard MCI:

- The affected task is canceled.
- R-transients write the machine check record to IJSYSRC, assess the damage, and continue system operation if possible.
- The system enters the hard wait state when a hard MCI:
  - Interrupts supervisor code execution
  - Occurs while critical information or phases from SYSRES are read
  - Damages privileged code by a permanent storage error
- The transients attempt to notify the operator about:
  - Machine check type
  - Wait state
  - Problem program termination
  - Buffer deletion

```
   Resident Machine Check Handler:

MACHEK   : Enqueue machine check into MC queue.
           Determine severity of the machine check.
           If MCIC shows 'CRW pending' then set
           corresponding indication in RAS linkage
           area (to retrieve and process CRW later-on).
           Handle uncorrected storage and storage key
           errors (Note 1).
           Indicate emergency handling in the RAS
           linkage area (if applicable), activate
           RAS task and return to interrupted task
           (load MC old PSW)
           or cancel damaged task              ----------> ERR1F
           (Note 2).

RASUPR   : RAS monitor (entered via task selection):
           If there is a machine check queued then
           check if $$RAST00 is already loaded in the
           RAS transient area (RTA), else use RAS fetch
           to load it, indicate machine check handling
           in the RAS linkage area and         ----------> $$RAST00
           If no error is indicated deactivate RAS task
           and exit to task selection          ----------> DISP

  RAS Transients

$$RAST00 : If system termination is indicated in the
           RAS Linkage Area, write hard wait message
           0T11W on SYSLOG and                 ----------> HARDWAIT
           If emergency handling is indicated, set
           system termination. If the recorder file
           is available                        ----------> $$RAST01
           else write hard wait message 0T11W
           on SYSLOG and                       ----------> HARDWAIT
           If it is not a terminating condition and
           the recorder file is available      ----------> $$RAST01
           else                                ----------> $$RAST03
           If a message is to be written (RASMSG field
           in the RAS monitor table (RASTAB))  ----------> $$RAST1 1
           Reset processing flags and set control
           register 14 and recording mode as required.
           Exit to RAS monitor                 ----------> RASSUPR

$$RAST01 : If the recorder file is not accessible
           and emergency handling indicated    ----------> $$RAST00
           else                                ----------> $$RAST03
```

Figure 7 (Part 1 of 3). Machine Check Handling

```
        If the recorder file is owned by EREP or
        ERP and emergency handling indicated  ----------> $$RAST00
        else wait until recorder file is available.
        Build interface segment (Note 3),
        set up machine check record and the
        record standard header in the
        RAS record area (RTACOM).
        If recorder file resides on CKD       ----------> $$RAST08
        If recorder file resides on FBA       ----------> $$RAST16

$$RAST08 : Read the recorder file header, check if
        the record(s) will fit in the file. Post message
        in the RASMSG field within the RASTAB if
        the threshold on recorder file is reached,
        the recorder file is full or an I/O error
        occurs during the recorder file access.
        Complete first record in the RAS record area (RTACOM)
        and write it on the recorder file. If indicated in
        the interface segment build additional records and
        write them on the recorder file.
        Update and rewrite the recorder file header record.
        If emergency processing, set the recording status
        in byte 2 of low storage (Note 1)
        and                           ----------> $$RAST00
        else                          ----------> $$RAST03

$$RAST16 : Read the recorder file header.
        Check if the record(s) will fit in the output
        buffer within the supervisor (buffer length is one
        FBA block). If it will fit, move record to the next
        available space in the buffer and write that block
        on the recorder file.If it does not fit,
        reformat block, move the record to the buffer,
        update the recorder file header and write buffer
        to the next block on the recorder file.
        When all records indicated by the interface segment
        are written to the recorder file and the header
        is updated during processing, rewrite recorder file
        header.
        If the threshold on the recorder file is
        reached, the recorder file is full or an
        I/O error occurs during recorder file access,
        post message in the RASMSG field within RASTAB.
        If emergency processing, set the recording status
        in byte 2 of low storage (Note 1)
        and                           ----------> $$RAST00
        else                          ----------> $$RAST03
```

Figure 7 (Part 2 of 3). Machine Check Handling

```
$$RAST03 : Check the machine check interruption code
        (MCIC) and set corresponding message bit in the
        RASMSG field within RASTAB (for uncorrected
        storage and storage key errors message request
        bits have already been set into the MC stack entry).
        If storage or storage key error messages have
        been requested                    ----------> $$RAST13
        if other message requested (in RASMSG) ---------> $$RAST11


$$RAST13 : According to the message request bits in MCST$MS1
        of the current MC stack entry
        issue following messages:
          0T15E MCAR REPAIR FAILED
          0T19E ALLOCATION OF XXR HAS BECOME INVALID.
                FAILING STORAGE ADDRESS XXXXXX.
          0T20E PFIX LIMIT REDUCED BY ONE PAGE
        Exit to                           ----------> $$RAST00

$$RAST11 : Write the following messages as indicated in
        the RASMSG field within RASTAB :
        0T00I THRESHOLD ON RECORDER FILE REACHED
        0T02I ERROR ON RECORDER FILE HEADER
        0T03I ERROR ON RECORDER FILE AT CCHHR
        0T05E RECORDER FILE FULL-RUN EREP
        0T08I C40 BUFFER PAGES DELETED=xxx
        0T09I SUCCESSFUL RECOVERY FROM MACHINE CHECK
        0T14E CLOCK DAMAGE. ALL MODES QUIET
        0T18E TIMER DAMAGE.
        0T21I SYSTEM PERFORMANCE DEGRADATION
        0T22E CLOCK AND OR TIMER DAMAGE
        0T25I SYSTEM RUNNING ON UPS
        0T26E UTILITY POWER RESTORED.
```

Figure 7 (Part 3 of 3). Machine Check Handling

Notes:

1. When a machine check indicating (uncorrected) storage or storage key error is raised, the affected storage block is validated with a TB (Test Block) instruction. If, in case of a storage key error, this validation is successful, the storage key will be reset.
   When the TB validation is not successful, messages 0T15 and 0T19 (if the FSA is within the ALLOCR area) or 0T20 (if the FSA is in a PFIX area) are scheduled via setting a message request bit in the MC stack entry.

   In either case the affected task is cancelled (ESA-mode: When the FSA belongs to a data space and the affected task is not the owner of that data space, the owning task is also cancelled).

2. The resident machine check handler analyzes the machine check interrupt code (MCIC) to see which of the following error types has occurred: System termination condition, hard machine check or recoverable machine check.

For **system termination condition**, the MCIC may indicate:

- No subclass bits on in bits 0-11 of MCIC.
- System damage.
- Instruction processing damage without backup condition.
- Warning bit on.
- Channel subsystem damage.
- Storage or storage key error (uncorrected) while the failing storage address is invalid.
- Storage or storage key error (uncorrected) where the FSA is within supervisor/SVA storage or the FSA is within partition storage and the storage validation failed .
- One or more old PSW bits are invalid (while the CPU is in the supervisor state).
- General registers, floating point registers or (ESA mode) access registers are invalid
- (370 mode) External damage with external damage code invalid.
- more than 10 machine checks are received within 10 minutes.

Action: Post C'A' in location 0 of low storage to assume 'Recording tried but unsuccessful'.
If a system task is affected then go to central hardwait
else indicate emergency handling in the RAS Linkage Area, post the RAS task to start recording and unpost the affected task.

With a **hard machine check** the system can continue but the damaged task is to be canceled. While the CPU is in the problem state, the MCIC indicates:

- The general registers, floating point registers or (ESA mode) access registers are invalid.
- A (uncorrected) storage or storage key error with FSA in partition storage and the software repair actions were successful.
- One or more old PSW bits are invalid.

Action: Activate RAS task and branch to the cancel routine in supervisor (ERR1F).

A **recoverable machine check** means the hardware recovered successfully from the error. None of the above conditions is present and only recording is required.

Action: Activate RAS system task (if not already active), reload registers from the MC save area and return to the interrupted code by loading the machine check old PSW.

3. $$RAST01 builds the interface segments located in the RAS monitor table (RASTAB). These interface segments are used to establish an interface with the record writer phases $$RAST08 and $$RAST16. They use this area to determine the number, contents, and lengths of the records to be built and recorded. The interface segment is 14 bytes long. Figure 8 on page 69 shows the format of the interface segment.

Displm		Length	Meaning
Dec	Hex		
0	0	4	Start address of machine check fixed logout area. Updated by the record writer phases to show the address of logout information to be included in the next record.
4	4	4	Start address of machine check or channel check extended logout area. Updated by the record writer phases to show the address of logout information to be included in the next record.
8	8	1	Sequence number of the next of n records to be entered. Not updated when last record is written.
9	9	1	Logout length for first record. Set to the maximum of 176 bytes unless the last record is written.
10	A	1	Record length. Set to the maximum of 200 bytes unless the last record is written.
11	B	1	Sequence number of last record to be written.
12	C	1	Logout length for last record.
13	D	1	Length of last record.

Figure 8. Interface Segment for RAS Record Writer Phases

Channel Check Handling

The supervisor resident channel check handler determines the severity of a channel- and interface-control check, allocates an error entry for the device, builds the error recovery information part within the error entry and chains the entry to the RAS error chain.

The channel check severity detect routine makes the decisions shown in Figure 9 on page 70.

Recovery actions for disk devices are handled by the supervisor resident channel check handler. For devices other than disk, the RAS task is posted and the recovery actions are done by the corresponding RAS transient.

The system is terminated if the channel error is unrecoverable and the channel user is a system task.

	Record error	Message on SYSLOG	Termi- nate System	CLRCH	HIO CLRIO	Recovery action
No ECSW stored		X	X			
Channel address invalid		X	X			
Unit address invalid	X	X			X	X
Inter- face in- operativ	X	X		X		X
System reset code on	X	X			X	X
CUA* valid	X	X				X
RECOVERY ACTION VERIFICATION						
	Retry channel program	Post error in CCB	Cancel channel user			
User own error recovery		X				
Channel program retryable	succ unsucc	X	X			
User accepts I/O error	succ unsucc	X				

Figure 9. Channel Check Handling Overview

```
  Resident Channel Check Handler:

CCENTRY1 : Determine severity of the channel check.
        If the system is to be terminated, post
        hard wait code in storage location zero,
        post RASSTERM in the RAS linkage area,
        enter $$RAST00 if it is loaded in the
        RAS transient area (RTA),          ----------> $$RAST00
        otherwise enter the wait state     ----------> HARDWAIT
        Allocate and build the device related error
        queue entry.
        If the device in error is a disk device,
        initialize retry of the failing channel
        program. On successful or unsuccessful retry
        or for non disk devices, enqueue the error
        entry in the RAS error chain, activate the
        RAS task.
        If unrecoverable error on disk device ----------> ERR1B
        else exit to task selection        ----------> DISP
        (Note 1)

RASUPR  : RAS monitor (entered via task selection):
        If the RAS error chain is empty, deactivate
        RAS task and exit to task selection  ----------> DISP
        If an active error entry is found on the
        RAS error chain, copy the error information
        to the work ERPIB and dequeue the entry
        from the RAS error chain.
        Check if $$RAST00 is already loaded in the
        RAS Transient Area (RTA), else use RAS fetch
        to load it, indicate channel check handling
        in the RAS linkage area and        ----------> $$RAST00

  RAS Transients

$$RAST00 : If system termination is indicated in the
        RAS linkage area, write hard wait message
        0T11W on SYSLOG and                ----------> HARDWAIT
        If emergency handling is indicated, set
        system termination. If the recorder file
        is available                       ----------> $$RAST02
        else write hard wait message 0T11W
        on SYSLOG and                      ----------> HARDWAIT
        If a channel check is to be handled  ----------> $$RAST02
        If a message is to be written (RASMSG field
        not zero)                          ----------> $$RAST11
```

Figure 10 (Part 1 of 6). Channel Check Handling

```
$$RAST00 : Reset processing flags.
(cont.)    Exit to RAS monitor                  ----------> RASSUPR

$$RAST02 : If recording is already done, dequeue
           the channel queue entry, cancel the task
           if user will not accept unrecoverable I/O
           errors, post the task ready from I/O wait
           and exit to                          ----------> $$RAST00
           If unrecoverable channel check, user own
           error handling or disk device is not
           indicated in the ERPIB, scan for and set
           the recovery transient id in the ERPID field
           in the RASTAB (Note 2).
           If the recorder file is not available and
           ERPID is set, exit to the indicated
           recovery transient                   ----------> $$RASTxx
           else exit to the message writer      ----------> $$RAST10
           Build channel check record in the RAS
           record area (RTACOM), using the information
           supplied by the ERPIB.               ----------> $$RAST01

$$RAST01 : If the recorder file is not accessable
           and emergency handling indicated     ----------> $$RAST00
           else load recovery phase indicated
           in the ERPID field if any,           ----------> $$RASTxx
           otherwise load message writer        ----------> $$RAST10
           If the recorder file is owned by ERP
           and emergency handling indicated     ----------> $$RAST00
           else wait until recorder file is available.
           Build interface segment (Figure 8 on page 69),
           insert OS device code in channel check record
           and build the record standard header
           within the RAS record area (RTACOM).
           If recorder file resides on CKD      ----------> $$RAST08
           If recorder file resides on FBA      ----------> $$RAST16
```

Figure 10 (Part 2 of 6). Channel Check Handling

```
$$RAST08 : Read the recorder file header, check if
           the record(s) will fit in the file. Post message
           in the RASMSG field in the RASTAB
           if the threshold on recorder file is reached,
           the recorder file is full or an I/O error
           occurs during the recorder file access.
           Complete first record in the RAS record area (RTACOM)
           and write it on the recorder file. If indicated in
           the interface segment build additional records and
           write them on the recorder file.
           Update and rewrite the recorder file header record.
           If emergency processing, set the recording status
           in byte 2 of low storage (Note 1)
           and                              ----------> $$RAST00
           else load recovery phase indicated
           in the ERPID field if any,       ----------> $$RASTxx
           otherwise load message writer    ----------> $$RAST10

$$RAST16 : Read the recorder file header.
           Check if the record(s) will fit in the output
           buffer within the supervisor (buffer length is one
           FBA block). If it will fit, move record to the next
           available space in the buffer and write that block
           on the recorder file.If it does not fit,
           reformat block, move the record to the buffer,
           update the recorder file header and write buffer
           to the next block on the recorder file.
           When all records indicated by the interface segment
           are written to the recorder file and the header
           is updated during processing, rewrite recorder file
           header.
           If the threshold on the recorder file
           is reached, the recorder file is full or an
           I/O error occurs during recorder file access,
           post message in the RASMSG field within the RASTAB.
           If emergency processing, set the recording status
           in byte 2 of low storage (Note 1)
           and                              ----------> $$RAST00
           else load recovery phase indicated
           in the ERPID field if any,       ----------> $$RASTxx
           otherwise load message writer    ----------> $$RAST10
```

Figure 10 (Part 3 of 6). Channel Check Handling

```
$$RAST05 : Recovery for 1050,1403,1442,1443,2501,2520
                2540,3210,3215,3800 and 3881
$$RAST06 : Recovery for 3505,3525,3540, and 3886
$$RAST07 : Recovery for 24xx tapes
$$RAST12 : Recovery for 3410, 3411, 3420, 3422, 3430,
                8809, 9346, 9347 tapes

        These transients have the logic as follows:
        Get CCB and CCW address from channel queue entry.
        If CCB or CCW address invalid or
        the sequence code in the ECSW is zero or
        the termination and sequence code indicates
        an unretryable error,
        post unrecoverable error message and
        cancel in the ERPIB and              ----------> $$RAST10
        If the failure occurs on SIO or
        the phase is called from $$RAST10 or
        the termination and sequence code indicates
        an retryable error,
        retry the channel program using the RAS monitor
        service (retry count for $$RAST07 and $$RAST12
        = 3). On successful retry post recovered
        channel error message and dequeue requested
        in the ERPIB and                     ----------> $$RAST10
        else post unrecoverable message and cancel
        in the ERPIB and                     ----------> $$RAST10
        If the termination and sequence code indicates
        a possible manual recovery, indicate channel error
        error and action message in the ERPIB ----------> $$RAST10
        For some termination and sequence codes the
        error will be ignored. In this cases channel
        error and dequeue is indicated in the
        ERPIB and                            ----------> $$RAST10
        (Figure 11 on page 79 and Figure 12 on page 81.)

$$RAST14 : VTAM owned phase
$$RAST15 : VTAM owned phase
$$RAST18 : 3890 support
$$RASTXA : Page-Printer support
```

Figure 10 (Part 4 of 6). Channel Check Handling

```
$$RAST19 : Recovery for 3424, 3480

        If I/O is retryable, request retry    ----------> RETRY
          from supervisor.
        Supervisor will return to $$RAST19    ---------
        If retry succeeds, set message code in ERPIB
          (MSGSEL) to "recovered" (RECCC).
          (RECCC).
        If retry fails, set MSGSEL to "unrecoverable"
          (HRDCC) and set status code in ERPIB (ERPIBSTC)
          to "cancel" (ERPIBCNC).

        If I/O is not retryable, set MSGSEL and ERPIBSTC
          as if retry failed, or leave them the same as
          they were upon entry to $$RAST19, depending on
          the reason they were unretryable.

        Request from supervisor transfer to    ---------> $$RAST10
          with MSGSEL and ERPIBSTC set as above

        Retryable means all the following are true:
          - Failing command CCW could not be found.  Some reasons
            for this:
            - CCB is not available
            - Channel program is not available
            - Supervisor did not provide failing CCW
              address in ERPIB
            - Channel program contains both command
              chaining and data chaining before failing
              CCW
            - Real address of a CCW could not be converted
              to virtual
          - Termination code is not 3
          - Unit Check was not in CSW status
          - CCW can be re-executed and achieve the same
            end result as if it had executed without
            error originally.  $$RAST19 makes this
            determination based on the command and the
            termination and sequence code from the ERPIB.
```

Figure 10 (Part 5 of 6). Channel Check Handling

```
$$RAST10 : Set up and write message indicated in the
           ERPIB:
            0T10I  CHANNEL ERROR RECOVERY ON cuu
            0T12I  IRRECOVERABLE CHANNEL ERROR ON cuu
            0T13A  CHANNEL ERROR ON cuu.
           For message 0T13A wait and analyze the
           operator response.
           If cancel is indicated by operator or by
           the ERPIB, reset the recovery transient
           identifier (ERPID) in the RAS monitor
           table (RASTAB) and              ----------> $$RAST02
           For a recovered channel check, dequeue the
           channel queue entry using the RAS monitor
           if indicated and                ----------> $$RAST00
           If the operator replies a retry, get the
           calling recovery phase from the ERPID field
           and return                      ----------> $$RASTxx

$$RAST11 : Write the following messages as indicated in
           the RASMSG field within RASTAB:
           0T00I THRESHOLD ON RECORDER FILE REACHED
           0T02I ERROR ON RECORDER FILE HEADER
           0T03I ERROR ON RECORDER FILE AT CCHHR
           0T05E RECORDER FILE FULL-RUN EREP
           Exit to                         ----------> $$RAST00
```

Figure 10 (Part 6 of 6). Channel Check Handling

Notes:

1. The resident CCH gains control from the I/O interrupt handler when either the interface control check or channel control check bit is posted in the CSW. The channel supplies additional channel check information in the 4-byte limited logout area (ECSW) and, under control of CR14 bit 2, in the I/O extended logout area. The ECSW is inspected to determine if enough information is valid to isolate the damage to either a channel or a device or if a system termination condition exists. For each channel check an error entry in the PUB extension is used to save error and recording information. If channel and device information is valid the error entry of the corresponding PUB is used. If a channel damage condition exist, the error entry of the first busy device not queued in error on the indicated channel is used.

   For channel checks on disk devices the recovery actions are initiated by the resident CCH. After recovery is done, the error entry is completed and chained to the RAS error chain. The RAS task is posted and control is given to the dispatcher. For an unsuccessful recovery the task in error is canceled.

   For channel checks on non-disk devices the error entry is completed, enqueued to the RAS error chain, the RAS task is posted and control is given to the dispatcher. Device dependent recovery actions and recovery dependent cancel actions are performed by the RAS monitor and the R-transients.

   When a system termination condition is detected, system termination is posted in the RAS linkage area and the RAS task is entered. The applicable termi-

nation code is posted at storage location 0.The following list gives the termination codes for the various types of disastrous channel errors:

```
C'B' = Irrecoverable channel check during fetch
C'C' = Irrecoverable channel check on paging channel
C'E' = ECSW not stored
C'G' = Channel error (invalid channel address)
C'H' = SYSLOG channel check while printing a MCAR or CCH message
```

Recording status code indicators in byte 2 of low storage are:

```
C'A' = Recording was tried but failed.
C'I' = Recording failed, but at least one record was written.
C'S' = Recording of all records was successful.
```

2. $$RAST02 uses an internal table (DEVTABLE) containing the VSE device code and the corresponding recovery transient identifier. If the VSE device code of the device in error is not contained in that table, unrecoverable channel error is posted in the ERPIB. The correspondence between device type and R-transient is given in Table 5.

Table 5. Correspondence of device type to recovery R-transient

DEVICE	PHASE
1050, 1403, 1442, 1443, 2501, 2520, 2540, 3210, 3215, 3800, 3881	$$RAST05
3505, 3525, 3540, 3886	$$RAST06
24xx tape	$$RAST07
3410, 3411, 3420, 3422, 3430, 8809, 9346, 9347 tape	$$RAST07
3424, 3480 tape	$$RAST19
3277, 3278, 3279 terminals. Supported as teleprocessing devices by VTAM.	$$RAST14
3705, 3791 control units. Supported as teleprocessing devices by VTAM.	$$RAST15
3890 (documented in 3890 support manual)	$$RAST18

Termination and Sequence Codes

The decisions taken by the channel check recovery transients depend on the termination and sequence codes passed to them in the ECSW in low storage and saved by the resident channel check handler in the ERPIB.

Termination Codes: These are 2 bit codes, given below in binary

```
00  =   Interface disconnect
01  =   Stop, stack, or normal termination
10  =   Selective reset
11  =   System reset
```

Sequence Codes: The sequence codes identify the I/O sequence in progress at the time of the error. They are meaningless if stored during the execution of HALT I/O or HALT DEVICE.

These are 3 bit codes and are given below in binary.

000 = Channel-detected error during the execution of a TEST I/O
001 = Command-out with a non-zero command byte, or bus-out was sent by the channel, but device status has not yet been analyzed. This code is set with a command-out to address-in during initial selection.
010 = Command accepted by the device, but no data has been transferred. Set by a service-out or command-out response to status-in during an initial selection sequence if the status posted in the channel status word (CSW) is
- channel end or
- channel end, device end or
- channel end, device end and status modifier or
- all zeros.

011 = At least one byte of data transferred over the interface. Set with a service-out response to service-in and, when proper, may be used when the channel is in an idle or polling state.
100 = The command in the current CCW has not yet been sent to the device or was sent but not accepted. Set for one of the following conditions:
- The command address is updated during command chaining or a START I/O.
- Service-out or command-out is raised in response to status-in during an initial selection sequence with the status on bus-in including attention, control unit end, unit check, unit exception, busy, status modifier (without channel end and device end), or device end (without channel end).
- When a short control-unit-busy sequence is signaled.
- When command retry is signaled.
- When the channel issues a TEST I/O rather than the command in the current CCW.

101 = Command accepted, but data transfer is unpredictable. Code applies from the time a device comes on the interface until it is found that a new sequence code applies. Used when a channel goes into polling or idle state and neither code two nor three can be found to apply. Also used when a channel cannot distinguish between code two or three.
110 and 111 = This error is handled as indicated in Figure 11 on page 79 and Figure 12 on page 81.

Operation	Term. Code	Sequence Codes 000	001	010	011	100	101	110	111
1403/1443 Printers									
Write (Only)		C	R	R	C	R	C	C	C
Write/Control	Not	C	R	C	C	R	C	C	C
Noop	applicable	C	R	R	R	R	R	C	R
Control		C	R	C	C	R	C	C	C
1442 Card Read Punch:									
Sense	00	C	R	R	R	R	R	C	R
	01	C	C	R	R	C	R	C	R
	10,11	C	R	C	C	R	C	C	C
Read/Control	00,10,11	C	R	C	C	R	C	C	C
	01	C	C	C	C	C	C	C	C
Write/Control	00,10,11	C	R	C	C	R	C	C	C
	01	C	C	C	C	R	C	C	C
2501 Card Reader:									
Read	00,10,11	C	R	A	A	R	A	C	A
	01	C	C	A	C	C	A	C	A
Sense	00,10,11	C	R	R	R	R	R	C	R
	01	C	C	R	R	C	R	C	R
2540 Card Read Punch and 3881 Optical Mark Reader:									
Read-Feed-SS		C	R	C	A	R	C	C	C
Read	Not	C	C	A	A	A	A	C	A
Feed-SS	applicable	C	A	C	C	A	C	C	C
Punch-Feed-SS		C	R	C	A	R	C	C	C
3210, 3215 Console Printer Keyboard:									
Read, Write	Not	C	R	R	C	R	C	C	C
Noop or Alarm	applicable	C	R	R	R	R	R	C	R

Note: R = retry, C = cancel, I = ignore, A = operator action

Figure 11 (Part 1 of 2). $$RAST05 Channel Check Recovery Decision Table

2520 Card Read/Punch: Operation	Term.Code	Sequence Codes: 000	001	010	011	100	101	110	111
Sense	00,10,11	C	R	R	R	R	R	C	R
	01	C	C	R	R	C	R	C	R
Read/Control	00,10,11	C	R	A	A	R	A	C	A
	01	C	C	A	A	C	A	C	A
Write/Control	00	C	R	A	A	R	A	C	A
	01	C	C	A	A	C	A	C	4
	10,11	C	R	R	R	R	R	C	R
Write, No Control	00	C	R	C	C	R	C	C	C
	01	C	C	C	C	C	C	C	C
	10,11	C	R	R	R	R	R	C	R
Immediate	00,10,11	C	R	C	C	R	C	C	C
	01	C	C	C	C	C	C	C	C
PRT1 Printer:									
Control	00,10,11	C	R	R	C	R	C	C	R
	01	C	R	C	C	R	C	C	C
Initializing or Diagnostic		C	R	R	R	R	R	C	R
Write/Control	01	C	R	C	C	R	C	C	C
	00,10,11	C	R	R	R	R	R	C	R
Write (Only)	01	C	R	R	C	R	C	C	C
	00,10,11	C	R	R	R	R	R	C	R
3800 Printing Subsystem:									
Forms	00,10,11	C	R	R	C	R	C	C	R
	01	C	R	R	C	R	C	C	C
Control, Load, and Status	01	C	R	R	R	R	R	C	R
	00,10,11	C	R	R	R	R	R	C	R
Write/Control	00,01	C	R	R	C	R	C	C	C
	10,11	C	R	R	R	R	R	C	R
Write (only)	all	C	R	R	R	R	R	C	R

Figure 11 (Part 2 of 2). $$RAST05 Channel Check Recovery Decision Table

Operation	Sequence Codes 000	001	010	011	100	101	110	111
3505, 3525 Card Devices :								
Test I/O	C	C	C	C	C	C	C	C
Feed—Stacker Select	C	R	A	C	R	C	C	C
Print	C	R	R	A	R	A	C	R
Write—Feed Stacker Select	C	R	R	A	R	A	C	R
Read—Feed Stacker Select	C	R	R	A	R	A	C	R
Other Commands	C	R	R	R	R	R	C	R
3540 Diskette Input/Output Unit:								
Read IPL	C	R	R	R	R	R	C	I
Feed	C	R	R	R	R	R	C	I
Define Ops	C	R	R	R	R	R	C	I
Sense	C	R	R	R	R	R	C	I
No—Op	C	R	R	R	R	R	C	I
Seek	C	R	R	R	R	R	C	I
Write Control	C	R	R	R	R	R	C	I
read Data	C	R	R	R	R	R	C	I
Write Data	C	R	R	R	R	R	C	I
Test I/O	C	C	C	C	C	C	C	C
Read	C	R	C	C	R	C	C	C
Eject	C	R	R	C	R	C	C	C
Scan,Noop,Load Line M.F. Load Page M.F.,Sense,L.For	C	R	R	R	R	R	C	R
Test I/O	C	C	C	C	C	C	C	C

Figure 12. $$RAST06 Channel Check Recovery Decision Table

Channel Report Word Handling

Channel Report Word (CRW) Handling is triggered whenever a machine check indicating 'CRW Pending' is raised. As a result of this machine check the resident machine check handler sets a 'CRW processing required' indication into the RAS linkage area.

The resident CRW handler (which acts as an appendage to the dispatcher) checks whether CRW processing is required. If yes, one (or more) CRWs are retrieved from the hardware and stored into the CRW queue for recording and message writing by the RAS transients.
In addition, the resident CRW handler performs special actions for the various CRW reporting sources (RSC) and error-recovery codes (ERC).

Table 6. CRW handling (Overview)

RSC	ERC	Handling	Message
Subchannel	Available	-	-
Subchannel	Installed Parameters Initialized	Update path masks and CHPID string in PUBX entry of affected subchannel. Set interrupt subclass 3 and enable subchannel. Call sense task path verification.	-
Subchannel	Installed Parameters Modified	Update path masks and CHPID string in PUBX entry of affected subchannel. Call sense task path verification.	-
Subchannel	Available	-	-
Channel Path	Terminal	Clear subchannel (CSCH) for all devices using the affected channel path and have a function pending. Reset path masks in all PUBX entries that are using the affected channel path. Reset the affected channel path (RCHP).	0T34
Channel Path	Permanent error with facility initialized	Update path masks and CHPID string in all PUBX entries that are using the affected channel path.	0T33
Channel Path	Permanent error with facility not initialized	Update path masks and CHPID string in all PUBX entries that are using the affected channel path.	0T33
Channel Path	Initialized	Update path masks and CHPID string in all PUBX entries that are using the affected channel path. Call sense task path verification.	-
Channel Path	Temporary error	-	0T31
Channel Subsystem	Event Information Pending	(in RAS transients) Retrieve the event information from the hardware and record it into the recorder file.	-
Configuration Alert Facility	Temporary Error	-	0T32
Monitoring Facility	Temporary Error	-	-

```
  Resident CRW handler:

MACHALLB (Dispatcher appendage in RAS for CRW handling):
    Retrieve CRWs from hardware (via STCRW instruction)
    and store them into a CRW queue element
    as long as CRWs are available.
    Process CRW reporting sources / error-recovery codes
    (see Table 6 on page 82).
    Post RAS task.
    Return to dispatcher.

RASUPR (RAS monitor, entered via task selection):
    If the CRW queue is not empty
    check whether $$RAST00 is already loaded in the
    RAS transient area (RTA), else use RAS fetch
    to load it, indicate 'CRW handling' in the
    RAS linkage area and                  ----------> $$RAST00

  RAS transients

$$RAST00: If called for CRW handling,
    fetch $$RAST01 and branch to it       ----------> $$RAST01

$$RAST01: If called for CRW handling,
    fetch $$RAST04 and branch to it       ----------> $$RAST04

$$RAST04: If called for CRW handling,
    then do
      - address first or next CRW in stack
      - build interface segment in RASTAB and header
        of CRW recorder file record
        -- recording of CRWs is not done when the
           'ancillary report bit' is set in the CRW
      - analyze CRW (or all CRWs in a chain) and
        -- schedule required message (see Table 6 on page 82)
           by appropriate bit settings in the CRW queue entry
        -- if CRW indicates 'event information pending'
           request processing of event information by
           $$RAST04 when called next time
      - if no CRW is to be recorded but a msg
        has to be issued                  ----------> $$RAST10
    If called for 'Event Information Pending',
    then do
      - retrieve event information from hardware
        via channel subsystem call
      - build interface segment in RASTAB and header
        of A2 recorder file record
    If recorder file on CKD               ----------> $$RAST08
    If recorder file on FBA               ----------> $$RAST16
```

Figure 13 (Part 1 of 2). CRW Handling

```
$$RAST08: If called for CRW recording, then do
     - for each CRW write a recorder file record
       -- do not record a CRW with 'ancillary report bit'
     - if CRWs are chained use recorder file record chaining
     - if any CRW related message is to be written  ------> $$RAST10
       else                                ----------> $$RAST00

$$RAST16: If called for CRW recording, then do
     - for each CRW write a recorder file record
       -- do not record a CRW with 'ancillary report bit'
     - if CRWs are chained use recorder file record chaining
     - if any CRW related message is to be written -------> $$RAST10
       else                                ----------> $$RAST00

$$RAST10: When requested via bit setting in the
     current CRW stack entry, issue one of the
     following messages to SYSLOG:
     0T30I CHANNEL REPORT WORD LOST DUE TO OVERFLOW CONDITION
     0T31I CHPID xx ALERT, UNSOLICITED MALFUNCTION INTERRUPT
     0T32I CHPID xx ALERT, NO ASSOCIATED SUBCHANNEL FOR DEVICE
     0T33I CHPID xx ALERT, CHANNEL PATH PERMANENT ERROR
     0T34I CHPID xx ALERT, CHANNEL PATH TERMINAL
     0T35I EVENT INFORMATION LOST DUE TO OVERFLOW CONDITION
```

Figure 13 (Part 2 of 2). CRW Handling

SVC44 Recording Interface

SVC44 is used for these functions:

- To write records from outside the A- or R-transient area
- To load a record builder phase
- To enqueue or dequeue the recorder file for use by EREP.
- To write BTAM-ES records.

Write from Outside the A- or R-Transient Area

Some phases (for example, $$BOESTV and $IJBSEOT) use SVC44 to request recording from outside the A- or R-transient areas. The record information is passed to SVC44 in register one and has the following layout :

```
Byte         0 : Length of record to be written
             1 : Record type code
             2 : Record dependent switch
                 Contains recording status on return
                   X'80' recording complete
                   X'10' record not written
             3 : Record dependent switch
       4 - 180 : Data to be recorded
```

For the contents of bytes 2 and 3 see “Data Areas” on page 89.

The four-byte interface area and the record address is saved in the ALTERNATE ERROR ENTRY by the resident SVC44 routine.

$$ABERA1 and $$ABERA3 (if the recorder file resides on CKD) read the recorder file header record and determine whether the record passed to it will fit in the file.If so, the record is written on the next available space in the file and the header record is updated and rewritten to the file. If the data portion consists of more than 176 bytes, only 176 are actually written.

$$ABERA9 (if the recorder file resides on FBA) reads the recorder file header record and determines whether the record passed to it will fit in the file. If so, the record is put in the next available space in the output buffer. The block is written on the file each time a record is added. Whenever a new block is allocated, the recorder file header record is updated and rewritten to the file. If the recorder file is not ready or is being accessed by EREP, exit is taken without recording.

Load a Record Builder Phase

To accomplish record building by an A-transient, the ROD or DVCDN command requests loading of that phase by SVC44. For this request, the interface area has the following layout :

```
Byte         0 : Length of information following byte 3
             1 : hex 'FF'
       2 -   3 : Last two characters of
                 transient name
             2 : Contains recording status on return
                   X'80' recording complete
                   X'10' record not written
       4 -   7 : PUB address of device to be handled
```

The proper A-transient phase is loaded, builds the record and calls the record writer phase $$ABERA1.

Enqueue or Dequeue the Recorder File

To prevent from accessing the recorder file while EREP is processing it, EREP requests enqueuing of the file via SVC44. In this case the interface area passed to SVC44 routine has the following layout :

```
Byte         0 : Zero
             1 : hex 'c5' if enqueue
                 hex 'c4' if dequeue
             2 : Zero
                 X'80' request handled on return
             3 : Zero
```

Enqueuing consists of posting bit 7 in RFFLAGS2 in the recorder file table and storing the key of the partition in which EREP is running. Dequeuing resets this indications.

BTAM-ES Recording

A BTAM-ES channel appendage routine may return to the I/O interrupt handler with a recording request. When the request is serviced, control is passed directly to $$ABERA5.

For a BTAM-ES error record, the first eight bytes of the error entry contain the following information:

```
1 byte   flags
1 byte   transmission count
1 byte   error count
1 byte   device type
4 bytes  terminal name
```

```
Supervisor resident routines :


SVC44    : Build ALTERNATE ERROR ENTRY, enqueue
           that entry in the ERP error chain and
           activate ERP-task.

ERP-task : Dequeue ALTERNATE ERROR ENTRY from the ERP
           error chain.
           If BTAM request                 ------------> $$ABERA5
           If record builder request (record type X'FF')
           load phase indicated in byte 6 and 7 of the
           error entry :
             DASD   -------->  BE
             3480   -------->  AY
             3800-1 -------->  BC,BD
             3800-3 -------->  XC,XD
             3890   -------->  AT
             3895   -------->  AR
             8809   -------->  AK
                                      ------------> $$ABERxx
           If record already build     ------------> $$ABERA6

ERPEXIT  : Deactivate ERP-task, return to requestor.



Transients loaded in the supervisor resident Physical Transient
Area (PTA) :


$$ABERA5   Build required record and       ------------> $$ABERA6

$$ABERxx   Build required record and       ------------> $$ABERA1
           If all records are processed, indicate
           recording status in the third byte of the
           interface area and              ------------> ERPEXIT

$$ABERA6 : Enqueue or dequeue recorder file if
           requested, indicate recording status
           in the third byte of the interface area -------> ERPEXIT
           Determine OS/VS device class and code
           and save it in the record.
           If ALTERNATE ENTRY processing and
           recorder file resides on CKD       ------------> $$ABERA3
           recorder file resides on FBA       ------------> $$ABERA9
           else                               ------------> $$ABERA1
```

Figure 14 (Part 1 of 2). SVC44 Recording

```
$$ABERA1:
        Build the 24 byte standard record header,
        write record on recorder file and update
        recorder file header if necessary   ----------> SVC 76
        If recording error                  ----------> $$ABERA2
        else return to caller indicated in the
        RFEXIT field                        ----------> $$ABERA5
                                                        $$ABERxx
                                                        ERPEXIT

$$ABERA3 and $$ABERA9 :
        Build the 24 byte standard record header,
        write record on recorder file and update
        recorder file header if necessary   ----------> SVC 76
        Indicate recording status in the third byte
        of the interface area.
        If recording error                  ----------> $$ABERA2
        else                                ----------> ERPEXIT

$$ABERA2 : Write indicated error message on SYSLOG :
        0T00I THRESHOLD ON RECORDER FILE REACHED
             (Last track of CKD recorder file or
              last 16 blocks of FBA recorder file ).
        0T02E ERROR IN RECORDER FILE HEADER
        0T03I ERROR ON RECORDER FILE AT CCHHR/PBN
        0T05I RECORDER FILE FULL--RUN EREP
        If called by $$ABERA3 or $$ABERA9  ----------> ERPEXIT
        else return to caller indicated in the
        RFEXIT field                        ----------> $$ABERA5
                                                        $$ABERxx
                                                        ERPEXIT
```

Figure 14 (Part 2 of 2). SVC44 Recording

Data Areas

This chapter contains information about the following data areas:

- Fixed Storage Locations in Processor Storage (Low Core)
- ERBLOC
- RAS Linkage Area (RASLINK)
- RAS Monitor Table (RASTAB)
- Error Recovery Procedure Information Block (ERPIB)
- Recorder File Table (RFTABLE)
- PUB2 Table
- Recorder File Record Formats

Fixed Storage Locations in Processor Storage (Low Core)

The allocation of the first 512 bytes of processor storage is standard for any IBM System/390* CPU. Fixed storage locations 513-1024 (X'200'-X'3FF) have been assigned to contain standard VSE Supervisor information. In Figure 15 the use of the fixed storage locations in processor storage are shown.

HEX	Label	Description
0– 7		Restart PSW if restart is possible (SDAID, DEBUG...)
0– 3		Hard wait message codes (MCH, CCH, IPL), if any
0– 4		Device error message codes if I/O error, and SYSLOG device is also in error.
10– 13		In a system with VTAM, the address of the VTAM communications vector table (ATCVT)
14– 17	CRADDR	Addr.of Communications Region of act. part.(COMREG)
18– 1F	EXOLDP	External Old PSW
20– 27	SVOLDP	Supervisor call old PSW
21	SVOLDKEY	Location of SVC old PSW key
24– 27	SVOLDADR	Address in SVC old PSW
28– 2F	PCOLDP	Program check old PSW
29	PCOLDKEY	Location of PC old PSW key
2C– 2F	PCKADR	Address in PC old PSW
30– 37	MCOLDP	Machine check old PSW
38– 3F	IOOPSW	I/O old PSW
40– 47	CSW	Channel status word
41– 43	CCWDRS	CSW channel command word (CCW) address
44	DEVSTA	Device status in CSW
45	CHNSTA	Channel status in CSW
46– 47	CSWCNT	Residual count
48– 4B	CAW	Channel address word
4C– 4F		Job duration
50– 53	TIMER	Hardware timer – no longer used
54– 57		Time of day – no longer used
58– 5F	XTNPSW	External new PSW
60– 67	SCNPSW	Supervisor call new PSW
64– 67	SCNADR	Address in SVC new PSW
68– 6F	PCNPSW	Program check new PSW
70– 77	MCNPSW	Machine check new PSW
78– 7F	IONPSW	I/O new PSW
80– 83	ASYSCOM	Address of System Communication Region (SYSCOM)
84– 85	EXTINF	External interrupt information
86– 87	EXTINF	External interrupt code
88– 89	SVCINF	SVC interrupt information
8A– 8B	SVCINTC	SVC interrupt code
8C– 8D	PGMINF	Program check interrupt information
8E– 8F	PGMINTC	Program check interrupt code

Figure 15 (Part 1 of 5). Fixed Storage Locations in Processor Storage

HEX	Label	Description	Bit
90– 93	TRADDR	Address which caused a page fault	
94– 95	MONCLASS	Monitor class	
96– 97	PERCODE	PER Code	
98– 9B	PERADDR	PER Address	
9C– 9F	MONCADR	Monitor call address field	
A0	ACCIDEXC	Access Register Id	
A1	ACCIDPER	Access Register Id (PER)	
A2– A7		reserved	
A8– AB		Target of STIDC instruction	
AC– AF	AIOEL	Extended I/O logout address	
B0– B3	EXCSW	Limited channel logout	
B0	ECSWDET	Error detector	
	ECSWSTAT	X'80' ECSW stored if bit=off	0
		X'40' Storage control unit (SCU) id	1
		X'20' dto.	2
	ECSWSCUS	X'10' SCU validity	3
	ECSWDCPU	X'08' Error detected by CPU	4
	ECSWDCHN	X'04' Error detected by channel	5
	ECSWDSCU	X'02' Error detected by SCU	6
	ECSWDSTO	X'01' Error detected by storage	7
B1	ECSWSRC	Source of error	
	ECSWSCPU	X'80' Source is CPU	8
	ECSWSCHN	X'40' Source is channel	9
	ECSWSSCU	X'20' Source is storage control	10
	ECSWSSTO	X'10' Source is storage	11
	ECSWSCNU	X'08' Source is control unit	12
		X'04' Reserved	13
		X'02' Reserved	14
	ECSWVLOG	X'01' Channel logout stored	15
B2	ECSWVAL	Field validity flag	
		X'80' Reserved	16
		X'40' Reserved	17
		X'20' Reserved	18
	ECSWVSEQ	X'10' Valid sequence code	19
	ECSWVUNS	X'08' Valid unit status	20
	ECSWVCAK	X'04' Valid CCW address and key	21
	ECSWVCHA	X'02' Valid channel address	22
	ECSWVUNA	X'01' Valid unit address	23

Figure 15 (Part 2 of 5). Fixed Storage Locations in Processor Storage

HEX	Label	Description	Bit
B3	ECSWTSC	Termination and sequence code	
		Bits 24 + 25 = Termination code	
		Interface disconnected — code: 00	
	ECSWTSER	X'80' Selective reset — code: 10	24
	ECSWTSSN	X'40' Stop, stack or normal term. — code: 01	25
	ECSWTSYR	X'C0' System reset — code: 11	
		X'20' Reserved	26
	ECSWTNOP	X'10' Interface inoperative	27
	ECSWTIOA	X'08' I/O error alert	28
		Bits 29 — 31 = Sequence code	
	ECSWTSC0	X'00' Error during TIO CLRIO — code: 000	
	ECSWTSC1	X'01' Command out status in error — code: 001	29
	ECSWTSC2	X'02' No data transfer — code: 010	30
	ECSWTSC3	X'03' Data transfer error — code: 011	
	ECSWTSC4	X'04' Command out not accepted — code: 100	31
	ECSWTSC5	X'05' Unpredicted data transfer — code: 101	
		X'06' Reserved — code: 110	
	ECSWTSC7	X'07' No other codes apply — code: 111	
B4— B7		Reserved	
B8— B9	IOINF	Saved I/O interrupt information	
BA— BB	CHNADR	I/O address	
BB	DEVADR	Device address	
BC— BF	INTPARM	XA I/O Int. Parameter	
C0- C3		Extended I/O Int. Info	
C4- D7		reserved	
D8- DF		CPU Time Save Area	
E0- E7		Clock Comparator Save Area	
E8— EF	MCIC	Machine check interruption code	
E8	MCICB00	MCIC byte 0	
	SDBIT	X'80' System damage	0
	PDBIT	X'40' Instruction processing damage	1
	SRBIT	X'20' System recovery	2
	TDBIT	X'10' Interval timer damage	3
	CDBIT	X'08' Timing facility damage	4
	EDBIT	X'04' External damage	5
		X'02' Unused	6
	DGBIT	X'01' Degradation	7
E9	MCICB01	MCIC byte 1	
	WABIT	X'80' Warning	8
	CPBIT	X'40' Channel report word pending	9
	SPBIT	X'20' Service Processor damage	10
	CKBIT	X'10' Channel Subsystem damage	11
		X'08' Unused	12
	VSBIT	X'04' Vector Facility Source	13
	BUBIT	X'02' Backed up	14
	DLBIT	X'01' Delayed	15
EA	MCICB02	MCIC byte 2	
	SEBIT	X'80' Storage error uncorrected	16
	SCBIT	X'40' Storage error corrected	17
	KEBIT	X'20' Storage key error uncorrected	18
	DSBIT	X'10' Storage degradation	19
	WPBIT	X'08' PSW EMPW validity	20
	MSBIT	X'04' PSW mask and key validity	21
	PMBIT	X'02' PSW program mask and condition code val.	22
	IABIT	X'01' PSW instruction address validity	23

Figure 15 (Part 3 of 5). Fixed Storage Locations in Processor Storage

HEX	Label	Description	Bit
EB	MCICB03	MCIC byte 3	
	FABIT	X'80' Failing storage address validity	24
	RCBIT	X'40' Region code validity	25
	EDRBIT	X'20' External damage code validity	26
	FPBIT	X'10' Floating point register validity	27
	GRBIT	X'08' General register validity	28
	CRBIT	X'04' Control register validity	29
	LGBIT	X'02' Log-out validity	30
	STBIT	X'01' Storage logical validity	31
EC	MCICB04	MCIC byte 4	
	IEBIT	X'80' Indirect Storage Error	32
	ARBITVAL	X'40' Access Register Validity	33
	DABIT	X'20' Delayed Access Exception	34
		X'10' Unused	35
		X'08' Unused	36
		X'04' Unused	37
		X'02' Unused	38
		X'01' Unused	39
ED	MCICB05	MCIC byte 5	
		X'80' Unused	40
		X'40' Unused	41
		X'20' Unused	42
		X'10' Unused	43
		X'08' Unused	44
		X'04' Unused	45
	CTBIT	X'02' CPU timer validity	46
	CCBIT	X'01' Clock comparator validity	47
EE- EF	MCELL	MC extended logout length Bits: 48-63	
F0- F3		Reserved	
F4	EDRCODE	External damage reason code	
		X'80' Reserved	
		X'40' Reserved	
	ESRBIT	X'20' Secondary report	
		X'10' Channel not operational	
		X'08' Channel control failure	
		X'04' I/O instruction time-out	
		X'02' I/O interruption time-out	
		X'01' Reserved	
F5- F7		Reserved	
F8- FB	FSA	Failing storage address	
FC- FF	REGCODE	Region Code (model dependent)	
100-1FF	FLOGA	Fixed logout area	
100-1FF		Store status or machine check save areas	
120-15F	ARSAVE	Access Registers 0 - 15	
160-17F	FPRSAVE	Floating point registers 0 - 6	
180-1BF	GRSAVE	General registers 0 - F	
1C0-1FF	CRSAVE	Control registers 0 - F	

Figure 15 (Part 4 of 5). Fixed Storage Locations in Processor Storage

HEX	Label	Description
200–203	IJBPATCH	Address of patch area
204–21F		CE patch area
220–223	ORB$	Address of ORB
224–227	IRB$	Address of IRB
228–22B	SCHIB$	Address of SCHIB
22C–237		reserved
238–239	XASUBSCH	Subchannel ..
23A–23B	XASUBSC1	.. number
23C–23D		reserved
23E–23F	CHNADRSA	CUU addr.from X'BA' at time system entered hard wait
240	SUPFLAG	Supervisor communication flag
	VMPMA	X'80' Running with PMA (under VM)
	PMRINIT	X'40' Page manager initialized
	SUPNFIX	X'20' Supervisor pageable
	VMSYS	X'10' System running under VM
	BFYSYS	X'08' BFY controlled processor
	TDACT	X'04' Turbo Dispatcher active
	KLLEDBT	X'02' BATCH deactivated
	TPBIT	X'01' TPIN in progress
241	SUPVFLAG	Supervisor internal flag
	JAACT	X'80' Job accounting active
	PBALACT	X'40' Partition balancing active
	TTIMEACT	X'20' Timer is active
	MPACT	X'10' More than one CPU active
		X'08' Reserved
	TTIMESET	X'04' Timer set
	XAACTIVE	X'02' XA/ESA capable
	TFREEPH	X'01' TFREE user phase area
242–243	RID	Routine identifier (RID)
244–247	ARUNTIME	Address of PCB which will be charged for accounting
248–24B	APIBTAB	Address of PIBAREA (PIB2TAB, PIBTAB)
24C–24F	ASYSPCB	Address of System PCB
250–253	ASCBATAB	Address of SCB address vector
254–257	SCBPTR	Address pf current SCB
258–25B	PSS	Partition selection string (PSS)
25C–25F	TDCPUPTR	Address of current CPU control block
260–263	TCBPTR	Address of currently active Task Control Block (TCB)
264–267	TIBPTR	Addr.of curr. active Task Information Block (TIB)
268–26B	PIBPTR2	Addr.of curr. active Part. Information Block2 (PIB2)
26C–26F	PCBPTR	Addr.of curr. active Partition Control Block (PCB)
270–273	XXARPTR	Address of DEBUG header
274–277		reserved
278–27B	AFLIH	Address of First Level Interrupt Handler (FLIH)
27C–27F	DISPAD	Address of Dispatcher
280–2BF	ERA	Save area for general registers 9 through 8
290–293	ERARD	Save area for general registers D
294–297	ERARE	Save area for general registers E
298–29B	ERARF	Save area for general registers F
29C–29F	ERAR0	Save area for general registers 0
2A0–2A3	ERAR1	Save area for general registers 1
2A4–2A7	ERAR2	Save area for general registers 2
2C0–2C3	ATIBATAB	Address of TIB table
2C4–2C7	APCBATAB	Address of Partition Priority Table (PPRTYOWN)
2C8–2CB	NPSQE	Number of available page frames
2CC–2CD	TINFRID	Save area for RID
2CE–2FF	SUPIDC1	Supervisor Level Identification information
300–3FF	SADUMPLA	Reserved area for Stand–alone DUMP, SDAID

Figure 15 (Part 5 of 5). Fixed Storage Locations in Processor Storage

ERBLOC Area

The ERBLOC is a an area in the SGATAB part of the supervisor used as an interface between system components involved in I/O ERP or Recording Request processing. The AERBLOC field in the System Communication Region (SYSCOM) contains a pointer to the ERBLOC. Internal macro ERBLOC maps the ERBLOC.

There are two formats of the ERBLOC:

- Regular Format, and
- Alternate Format.

Regular Format is used for ERP (processing error conditions) and Alternate Format is used for processing Logging Requests.

Bytes Dec	Bytes Hex	Label	Description
0-7	0-7	SVC5NM	Name of first/next ERP Transient to be fetched
8-11	8-B	YRETRY	Continuation address for retry I/O request (addr of INITRG)
12-15	C-F	YIGNORE	Continuation address to ignore I/O error (addr of IGNORE)
16-19	10-13	ACANCEL	Continuation address to cancel I/O request (addr of ERR1A)
20-23	14-17	YERPEXIT	Common DSK/ERP return address (addr of ERPEXIT)
24-307	18-133		Varies--see below
308-323	134-143	ERCHNOFT	Chain header offset table, used to select one of 4 error entry chains based on bits in error block and priority scheme
324-327	144-147	RASERCHN	Addr of 1st error block on RAS chain
328-331	148-14B		Pointer to RAS TIB
332-335	14C-14F	ERPERCHN	Addr of 1st error block on ERP chain
336-339	150-153		Pointer to ERP TIB
340-343	154-157	DSKERCHN	Addr of 1st error block on DSK chain
344-347	158-15B		Pointer to DSK TIB
348-351	15C-15F	SNSERCHN	Addr of 1st error block on SNS chain
352-355	160-163		Pointer to SNS TIB

Figure 16. ERBLOC

Bytes Dec	Bytes Hex	Label	Description
24-31	18-1F	ERQCSW	CSW from I/O interrupt or SIOF (Exception: see Note 1)
32-33	20-21	ERQPUB	PUB entry address for device with error condition
34	22	ERQFLG	Flag Byte. Bits defined below
		TRUNRF	Bit 0: No record found on DASD
		OPINT	Bit 1: Device should be set in intervention required state (set by transients)
		PASSBK	Bit 2: Error info should be passed back to I/O requester (set by transients)
		IGNERR	Bit 3: Transients may request IGNORE processing from supervisor (set by transients)
		SUCCESS	Bit 4: No I/O error exists (either was recovered or never actually existed). (set by transients)
		RTYERR	Bit 5: Transients may request RETRY processing from supervisor (set by transients)
		RECONLY	Bit 6: Only recording is needed for this error condition. Set by supervisor.
		OCCUP	Bit 7: ERBLOC is in use. Always 1 while A-transient running.
35	23	ERQMSG	Message code for message that is to be issued. Format: unsigned packed decimal coding of 'xx' in '0Pxx'. Also has value x'E2' to indicate a 'soft' disk error. This field has the same location as ERQAEMSG in the alternate-format ERBLOC. ERQMSG and ERQAEMSG have disjoint sets of possible values and hence this field can be used to determine the ERBLOC format. This field is x'20' at entry to transient to indicate the Sense failed.
36-39	24-27	ERQSEK	Used for disk devices only CKD: Failing Seek address FBA: OS device type codes

Figure 17 (Part 1 of 2). Bytes x'18' - x'63' of Regular-format ERBLOC

40-43	28-2B	ERQCCB	ERQCHQPT and ERQCCBA
40	28	ERQCHQPT	Index of channel queue entry X'FF' for unsolicited interrupt or intercept error condition
41-43	29-2B		addr of CCB (0 if I/O request has been posted complete or unsolicited interrupt.
44-75	2C-4B	ERQSNS	Sense data if error condition is UC. All zero if not UC. If device gave less than 32 bytes, rest are 0.
76-79	4C-4F	ERQPUB2A	Addr of PUB2 entry for device with error condition
80-83	50-53	ERQCAW	CAW (key/CCW address) that supervisor should use for retry I/O or 0 if supervisor should compute its own CAW and do channel program prefixing if necessary. Set by transients
84-95	54-5B	ERQPRINF	Error processing information
84	54	ERQLSNS	Number of sense bytes device gave
85	55	ERQPRFL	Flags. Bits defined below
		ERQPSOV	Bit 0: More than 1 Unit Check
		ERQPVT	Bit 1: Path Verification required Bit 2: reserved
		ERQV510	Bit 3: Extended ERBLOC Version
		ERQPMSG	Bit 4: Only message writer needed to process error condition. Bits 5-7: reserved
86	56	ERQERPF	supvr/transient communication flags Bit 0: Reserved
		ERQPOBR	Bit 1: OBR must be recorded
		ERQPMDR	Bit 2: MDR must be recorded
		ERQPSIR	Bit 3: Alert record must be recorded
		ERQPPER	Bit 4: Permanent error has occurred
		ERQPTEM	Bit 5: Temporary error has occurred
		ERQPSOF	Bit 6: Soft error has occurred
		ERQNREC	Bit 7: No recording should be done
87	57	ERQPATH	Path failure information
88-91	58-5B	ERQCOMM	ERP communications area
92-95	5C-5F		Reserved
96	60	ERQACPU	CPU failure information
97	61	ERQRSV	Reserved
98	62	ERQERR	Error count
99	63	ERQCCODE	Command code set by ERP
100-103	64-67	ERQCHQA	Reserved
104-111	68-6F	ERQFJOB	Failing job name
112-115	70-73	ERQSCSW	Subchannel status word 0
116-119	74-77	ERQESW	Extended Status Word 0
120-123	78-7B	ERQPUBNO	PUB Index
124-125	7C-7D	ERQTID	Task Id of failing task
126-127	7E-7F	ERQRFRLN	Length of recording area
128-131	80-83	ERQFRAD	Address of recording area
132-307	84-133	ERQFREC	RF Record Area

Figure 17 (Part 2 of 2). Bytes x'18' - x'63' of Regular-format ERBLOC

Notes:

1. The ERQCSW field is not necessarily a CSW constructed by the channel. When the error condition is a Condition Code 3 (immediate or deferred) from a SIOF, an unsolicited interrupt, or an intercept condition, the supervisor makes up the CSW itself. In these cases, the status bytes in ERQCSW are x'00FE'. In the Condition Code 3 case, the key field in ERQCSW is the key field from the CAW, the Suspend, Logout, and Condition Code and Residual Count fields are 0, and the CCW address is the CAW CCW address + 8. In the other two cases, all fields are 0 except the status fields.

   In some cases, the supervisor adds CE and DE bits to the status presented by the device.

   The first byte of ERQCSW is never that set by the channel, but rather the first byte of the CAW that VSE used (except in the special cases described above).

Bytes Dec	Bytes Hex	Label	Description
24-27	18-1B		varies--see below
32-33	20-21	ERQAEPUB	Addr of PUB entry for affected device. Not defined for record-built SVC 44 requests
34	22	ERQAEFLG	Flags. Bits defined below
		ERQAEFL0	Bit 0: SDR is TFIXed
		ERQAEFL6	Bit 6: 0
		ERQAEFL7	Bit 7: Error block is in use. Always 1 when A-transient running. Note that this is identical to the OCCUP bit in the regular format ERBLOC and in the same place.
35	23	ERQAEMSG	x'AE'. Because this value is impossible for ERQMSG, which occupies this entry in the regular format ERBLOC, this field can be used to determine the format of the ERBLOC.
36-39	24-27	ERQAETIB	TIB of requesting task
40-43	28-2B		Reserved
44-75	2C-4B	ERQAECOM	Communication information
76-79	4C-4F	ERQAEP2A	Addr of PUB2 entry for affected device
80-91	50-5B		Reserved
92-95	5C-5F	ERQAEPBX	Addr of PUBX entry for affected device
96-97	60-61		Reserved
98	62	ERQAEERR	Error Count
99	63		Reserved
100-103	64-67	ERQAECHQ	Address of Chanq Entry
104-111	68-6F	ERQAEJOB	Failing job name
112-119	70-77	ERQAESRV	Reserved
120-125	78-7D		Reserved
126-127	7E-7F	ERQAERLN	Length of recording area
128-131	80-83	ERQAERAD	Address of recording area
132-307	84-133	ERQAEBUF	RF Record Area

Figure 18. Bytes x'18' - x'133' of Alternate-format ERBLOC

24-27	18-1B	ERQAEADR	addr of SVC 44 parameter list. Even if Recording Request is not from SVC 44, there is an SVC 44 parameter list.
28-31	1C-1F	ERQAEINF	Logging Request information
28	1C	ERQAELEN	Length of SD record or if record builder request, length of SVC 44 parameter list minus 4.
29	1D	ERQAETYP	Record type of SDR to be recorded. If record builder request, x'FF' (A-transients decide record type).
30	1E	ERQAESW1	Record dependent switch 1--non-record builder request only
31	1F	ERQAESW2	Record dependent switch 2--non-record builder request only
30-31	1E-1F		Last 2 characters of name of 1st A-transient to processing logging request (record builder request only)

Figure 19. Bytes x'18' - x'1F' of Alt ERBLOC except for BTAM-ES request

24	18		Flags
25	19	BTAMSIO	SIO count
26	1A	BTAMERR	error count
27	1B	BTAMDVT	device type
28-31	1C-1F	BTAMTERM	terminal name

Figure 20. Bytes x'18' - x'1F' of Alt ERBLOC for BTAM-ES request

Machine and Channel Check Control Blocks

RAS Linkage Area (RASLINK)
RAS Monitor Table (RASTAB)
Error Recovery Procedure Information Block (ERPIB)

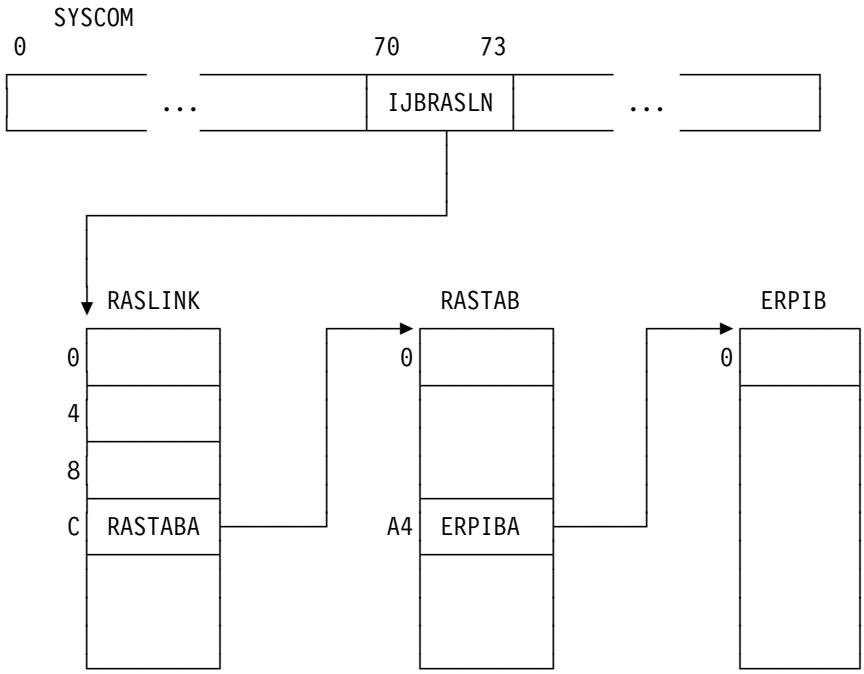

Figure 21. Machine/Channel Check Control Block Relationship

RAS Linkage Area (RASLINK)

Dec	Hex	Label	Description
0–3	0–3	CPUIDW1	First part of CPUID field
4–7	4–7	CPUIDW2	Second part of CPUID field
5	5	CPUID	Model number in CPUID field
6	6	RASMCELL	Length of machine check extended logout area
8	8	RASDMC	Damaged channel ID
9	9	RASFLAGS	RAS flag byte
		RASACT	X'80' RAS task activated
		RASMCACT	40 Machine check handling
		RASCCACT	20 Channel check handling
		RASCRWAC	10 CRW Handling in Progress
		RASEMGEX	08 Emergency Handling
		RASSTERM	04 Terminate System
			02 reserved
		RTAIOA	01 RAS task I/O active
10	A	MCFLAGS	Machine check flags
		MCHARD	X'80' Hard machine check
		MCEXTD	X'08' External damage to process
		MCCRWP	X'04' CRW Pending Req. to process
		MCEVIP	X'02' Event Information Pending
11	B	RASRSFLG	RAS recording status flag
		RASDEBUG	X'80' RAS Debug Mode
			40 Reserved
		RASNOMSG	20 Unrecoverable channel check on SYSLOG
			10 Reserved
			08 Reserved
		RASBTDEQ	04 BTAM dequeue request
		RASMSGRT	02 Return from RAS message writer
		RASMSGIO	01 RAS message I/O
12–15	C–F	RASTABA	Address of RAS monitor table (RASTAB)
16–19	10–13	RASBASE	RAS base address
20–21	14–15	RASIMOD	Internal model number
22–23	16–17	RASIOELL	Length of I/O extended logout area
24–27	18–1B	RASMCELA	Address of machine check extended logout area
			X'80' Indicates field contents not valid
28–31	1C–1F	RASTAREA	Address of RAS Transient Area
32–35	20–23	RAS$PUBX	A(address of PUBX area)
36–47	24–2F	RASPGRID	Path Group ID (IPL)
48	30	RASCPVER	Real CPU Version Code (SPDT)
49–51	31–33		reserved
52–55	34–37	RASCHCB	Ch Subsys ContrBlock Address
56–59	38–3B	RASMCSTK	Mach Check Stack Area Address

Figure 22. RAS Linkage Area (RASLINK)

RAS Monitor Table (RASTAB)

Bytes Dec	Bytes Hex	Label	Description
0–3	0–3	RASMCERQ	Machine Check error queue
4–7	4–7	RASCRWQE	CRW entry currently processed
8–103	8–67		reserved
104–115	68–77	RASCCB	RAS CCB
116–147	78–97	RASCCWS	RAS CCW chain
148–154	98–9E	RASEEK	Seek address of RAS seek
155	9F	RTAOWN	R–transient identifier
156–159	A0–A3		reserved
160–163	A4–A7	ERPIBA	Address of work ERPIB
164–167	A8–AB		reserved
168	AC	RTAID	Requestor ID for RTA I/O
		RASRECID	X'08' RAS recording request
		RASRTYID	X'04' Channel retry request
169	AD	ERPID	Return load index for WTOR
170–171	AE–AF	RASRES	Device address of SYSRES
172–173	B0–B1	RASREC	Device address of SYSREC
174–175	B2–B3	RASLOG	Device address of SYSLOG
176–243	B4–F3	TRANSAV	RTA register save area, Register 0 to Register 15
244–307	F4–133	SYSREGS	RAS monitor register save area, Register 0 to Register 15
308–311	134–137	SUPLINK	Service routine address for RTA in RAS monitor
308	134	LINKFLAG	Flag byte indicating requested service
		RASLIO	X'80' Perform normal I/O
		RASLEMIO	40 Perform emergency I/O
		RASLFTCH	20 Fetch another transient
		RASLWAIT	10 Perform wait
		RASLPDEQ	08 Dequeue page frame
		RASLDEQ	04 Dequeue CCB/IORB
		RASLFREE	02 Free I/O extended logout area
			01 reserved
		RASLEXIT	00 Exit from RAS transient
312–323	138–13F	HIR	Hardware instr. retry accumulator
312–313	138–139	HIRACNT	Accumulated HIR count
314–315	13A–13B	HIRLCNT	Threshold value for count
316–319	13C–13F	HIR1TME	Time of day for first error of group
320–323	140–143	HIRLTME	Time threshold value in timer units
324–335	144–14F	ECCMAIN	Main storage error accumulators
324–325	144–145	ECCACNT	Accumulated ECC count for main stor.
326–327	146–147	ECCLCNT	Threshold value for count
328–331	148–14B	ECC1TME	Time of day for first error of group
332–335	14C–14F	ECCLTME	Time threshold value in timer units

Figure 23 (Part 1 of 2). RAS Monitor Table (RASTAB)

Bytes Dec	Bytes Hex	Label	Description
336–347	150–15B	MACHERR	Machine check accumulators
336–337	150–151	MCERRCNT	Accumulated MCH check count
338–339	152–153	MCLCNT	Threshold value for count
340–343	154–157	MC1TIME	Time of day for first error of group
344–347	158–15B	MCLTIME	Time threshold value in timer units
348	15C	MCMODE	Hardware operation mode
349	15D	BUFDEL	Count of buffers deleted
350	15E	RASMSG1	RAS Message byte 1
		MCPR	X'80' POWER Restored — running normal
		MCPL	40 POWER Lost — running on UPS
			20 reserved
		MTICLDMG	10 Clock and or timer damage
		MTIMDMG	08 Timer damage
		MECQUIET	04 Control storage ECC in quiet mode
		MPERFDEG	02 System performance degradation
			01 reserved
351	15F	RASMSG2	RAS Message byte 2
		MCLOKDMG	X'80' Clock damage, all modes quiet
		MLASTTR	40 Threshold on recorder file reached
		MPAGEDEL	20 Buffer pages deleted
			10 reserved
			08 reserved
		MFILEFL	04 Recorder file full
		MUNRCIO	02 Error on recorder file
		MCRECOV	01 Successful recovery from machine check
352	160	RASMSG3	RAS Message byte 3 (reserved)
353–355	161–163		reserved
356–357	164–165	RASIND	RAS indicators
		RASNODEQ	X'80' Page frame not dequeued
358–363	166–16B		Reserved
364–376	16C–178	INTERSEG	Interface segment build area
364–367	16C–16F	FXDLGADD	Address of fixed logout
368–371	170–173	EXTLGADD	Address of extended logout
372	174	INOFN	Sequence number record one of n
373	175	ILOGL	Logout length in record one
374	176	IRECL	Total length of record one
375	177	NNOFN	Sequence number record n of n
376	178	NLOGL	Logout length in record n
377	179	NRECL	Total length of record n
378–379	17A–17B		reserved
380–384	17C–17F	MACHIND	Indicate Machine Check

Figure 23 (Part 2 of 2). RAS Monitor Table (RASTAB)

Error Recovery Procedure Information Block (ERPIB)

Bytes Dec	Bytes Hex	Label	Description
0–7	0–7	ERPIBCSW	Saved CSW
0	0	ERPIBSTC	ERPIB status codes
		ERPIBEND	X'FF' Indicate end of ERPIB
		ERPIBFRE	X'FE' Indicate free ERPIB
		ERPIBCNC	X'FD' Indicate task is to be canceled
		ERPIBCCR	X'FC' Indicate retry unsuccessful
		ERPIBCCS	X'FB' Indicate retry successful
0–3	0–3	ERPIBCCW	Address of failing CCW + 8
4	4	ERPIBST1	First status byte
5	5	ERPIBST2	Second status byte
6–7	6–7	ERPIBCNT	Residual count in CSW
8–11	8–B	ERPIBIOE	Pointer to corresponding I/O extended logout area
12	C		Reserved
13	D	ERPIBDMC	Damaged channel ID
14–15	E–F	ERPIBPUB	PUB address of failing device
16	10	ERPIBCQP	Channel queue pointer from the PUB
17	11	ERPIBRTC	RAS retry counter
18	12	ERPIBMSG	Message indicator
		ACTMSG	X'80' Wait for operator response
		CCDONE	40 Channel check handling complete
		CCNODEQ	20 PUB not queued in error
			10 Reserved
			08 Reserved
			04 Reserved
		RECCC	03 Recovered channel check
		ERRCC	02 Channel check
		HRDCC	01 Unrecoverable channel check
19	13	ERPIBREQ	Requestor ID
20	14	ERPIBFLG	Flag byte
		CCSIO	X'80' Channel check on SIO
		CCDAM	40 Channel damage
			20 Reserved
			10 Reserved
		CCREC	08 Record build or written
			04 Reserved
		CCDSK	02 Channel check on disk device
		CCSKM	01 Skip message writer
21–23	15–17	ERPIBESW	Extended CSW
23	17	ERPIBSC	ERPIB Sequence Code
23	17	ERPIBTCD	ERPIB Termination Code
		CCAREP	X'20' Chnl Check Ancillary Report
24–27	18–1B		reserved
28–31	1C–1F	ERPIBCQA	Chanq Address
32–39	20–27	ERPIBTOD	TOD Clock
40–42	28–29		reserved

Figure 24. Error Recovery Procedure Information Block (ERPIB)

Recorder File Table (RFTABLE)

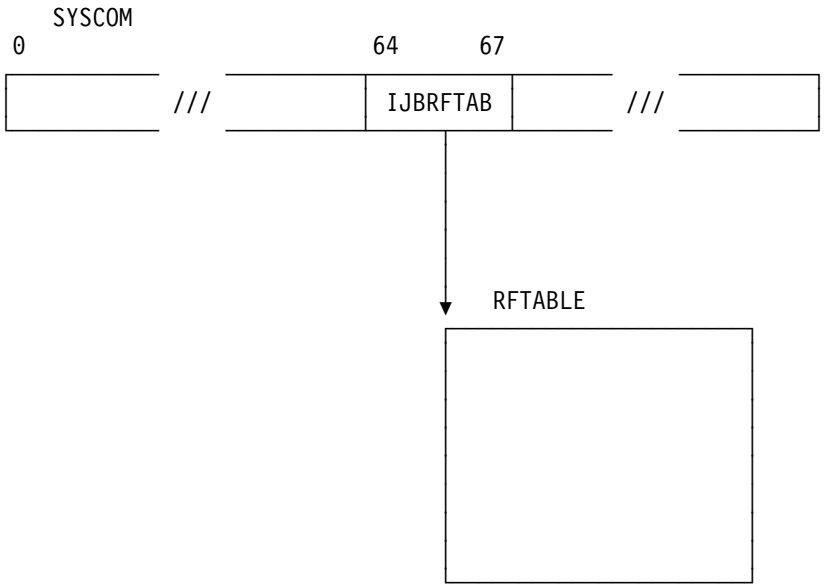

Figure 25. Recorder File Table Relationship

Bytes Dec	Bytes Hex	Label	Description
		RFTABLE	Label of Starting Address
0	0	RFFLAGS1	Flag byte 1
		RFFULL	X'80' File full
		RFRDE	40 RDE option included
		RFIPL	20 Initial IPL
		RFNO	10 RF=No option
		RFCREATE	08 File is to be created
		RFBUILT	04 File has been created
		RFONFBA	02 File on FBA device
		RFREADY	01 File ready
1	1	RFFLAGS2	Flag byte 2
		FFMSG	X'80' File full message request
		LTMSG	40 Last track message request
		IEMSG	20 I/O error message request
		DLMSG	10 Data lost message request
		RFEVA	08 EVA message request
		RFRTAOWN	04 File owned by RTA recorder
		RFPTAOWN	02 File owned by PTA recorder
		RFEREP	01 File being accessed by EREP
2	2	RFFLAGS3	Flag byte 3
		LTMISSUD	X'80' Last track msg issued once
		RECDERR	40 Error is to be recorded
		RECDSF	20 Short form record request
		RFIRULT	10 Individual records for unlabeled tapes
		FFMSG1ST	08 First time file full
		RFHIOERR	04 Error in writing RFHEADER
		RFBOMT05	02 Exit to $$BOMT05 indicator for $$BOPEN
		RFBOMT01	01 Exit to $$BOMT01 indicator for $$BOPEN
3	3	RFFLAGS4	Flag byte 4
			X'80' — X'02' Reserved
		RFRNW	01 No record written
4	4	RFFLAGS5	Flag byte 5
			X'80' — X'02' Reserved
		RFFLG5BD	01 BOPEND called by OPEN
5	5	RFNOFN	N of N for OBR records (low order 4 bits contain the number of records to be recorded and high order 4 bits contain the number of the record being recorded)
6	6	RFRECTYP	Record type code
7	7	RFREL	Release level code of VSE/Adv.Funct.
8	8	RFRDSW1	Record dependent bit 1
		RFTEMP	X'40' Temporary error
		RFALTPIO	X'10' Retried on alternate path

Figure 26 (Part 1 of 2). Recorder File Table (RFTABLE)

Bytes Dec	Bytes Hex	Label	Description
9	9	RFRDSW2	Record dependent bit 2
10 – 11	A – B	RFBUFLG	Length of data buffer
12 – 15	C – F	RFBUFAD	Address of data buffer
16 – 17	10 – 11	RFNAVR	Offset to current RDF
18 – 19	12 – 13	RFRECLN	Length of record
20	14	RFRDSW3	Rec dependent switch 3
21	15		Reserved
22 – 23	16 – 17	RFRCLCKD	CKD Block Length
24 – 27	18 – 1B	RFRECADR	Address of record
..... CKD Device Related Information			
28 – 34	1C – 22	RFSEEK	Work area for seek addr.BBCCHHR
28 – 29	1C – 1D	RFSEEKBB	BB portion of seek
30 – 31	1E – 1F	RFSEEKCC	CC portion of seek
32 – 33	20 – 21	RFSEEKHH	HH portion of seek
34	22	RFSEEKR	R portion of seek
35	23		Reserved
36 – 39	24 – 27	RFHDRCH	SYSREC cylinder/head
36 – 37	24 – 25	RFHDRCYL	Cyl. address of file start
38 – 39	26 – 27	RFHDRTRK	Head address of file start
..... End of CKD Device Related Information			
..... FBA Device Related Information			
28 – 31	1C – 1F	RFCUBL	Work area for block number
32 – 35	20 – 23		Reserved
36 – 39	24 – 27	RFHDRBL	SYSREC block number
..... End of FBA Device Related Information			
40 – 41	28 – 29	RFCHMAP	Map of supported channels
42 – 49	2A – 31	RFCHIDC	Channel ID codes
50 – 51	32 – 33	RFEREPID	EREP taskid for EOTSK
52 – 55	34 – 37	RFEXIT	Exit phase name or exit address
52 – 53	34 – 35	RFEXITPH	Phase Id to be fetched
52	34	RFEXIT0	Name or Exit Addr Byte 0
53	35	RFEXIT1	Name or Exit Addr Byte 1
56	38	RFEVARTH	EVA read threshold
57	39	RFEVAWTH	EVA write threshold
58 – 59	3A – 3B	RFP2ENTL	Length of PUB2 table
60 – 63	3C – 3F	RFP2ENT	Address of PUB2 table
64 – ...	40 – ...	RFP2ITAB	PUB2 index table (see Note)

Note: Two bytes are generated for each PUB2 index entry.

RFP2ENTL, RFP2ENT, RFP2ITAB are only set and valid when the supervisor is NOT generated for 'More Devices'.

Figure 26 (Part 2 of 2). Recorder File Table (RFTABLE)

Physical Unit Block Table 2 (PUB2)

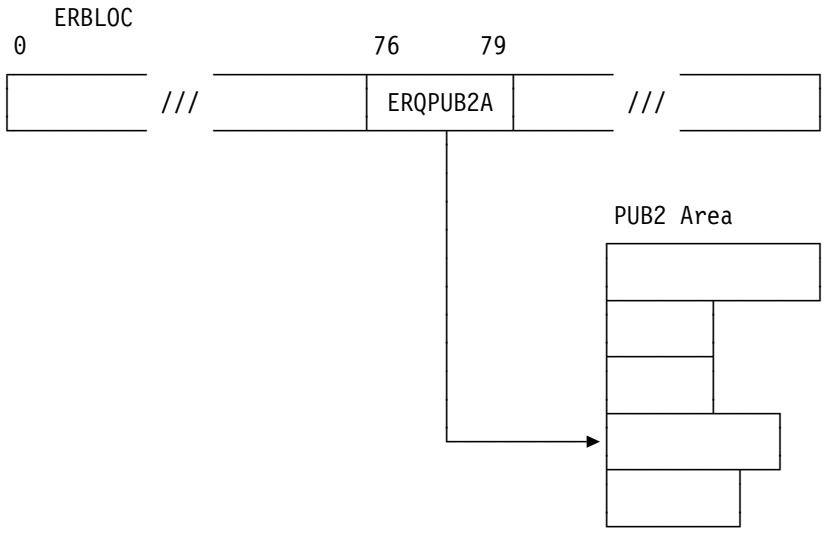

Figure 27. PUB2 Relationship

Bytes Dec	Bytes Hex	Label	Description
0 – 2	0 – 2	P2USAGE	Usage count (number of non–ERP SIO)
3	3	P2FLAGS	Flag byte common to all PUB2 entries
		P2INTSM	X'80' Device is in intensive mode
		P2DIAGM	40 Device is in diagnostic mode
		P2NORCM	20 No recording mode
		P2STAT2	10 Call statistics transient 2
		P2NAMEF	08 Use PUB2 name completion field
		P2OPEN	04 Volume opened on this device
		P2NOSIO	02 Intercept next SIO
		P2DOMSV	01 Indicate delete operator msg.
4	4	P2LIMIT	CE mode limit byte
5	5	P2BBMASK	CE mode byte/bit mask
6–	6–	PUB2EXT	PUB2 extension. This is defined differently for each kind of device. The length is also dependant on the kind of device.

Figure 28. Physical Unit Block Table 2 (PUB2)

Physical Unit Block Table 2 Extensions

Bytes Dec	Bytes Hex	Label	Description
6	6	P2UNITX	Start of unit record PUB2
6 – 11	6 – B	SDRUNITR	SDR counters for unit record devices
12	C	P2UNITE	End of unit record PUB2

Figure 29. Unit Record and Unsupported Device PUB2 Extension

Bytes Dec	Bytes Hex	Label	Description
6	6	P23540X	Start of PUB2 extension
6 – 13	6 – D	SDR3540	SDR counters
14 – 15	E – F	P23540R	Reserved
16	10	P23540E	End of 3540 PUB2

Figure 30. 3540 Diskette PUB2 Extension

Bytes Dec	Bytes Hex	Label	Description
6	6	P23211X	Start of PUB2 extension
6 – 11	6 – B	SDR3211	SDR counter area
12	C	P23211E	End of 3211 PUB2

Figure 31. 3211 Printer PUB2 Extension

Bytes Dec	Bytes Hex	Label	Description
6	6	P24248X	Start of PUB2 extension
6 – 11	6 – B	SDR4248	SDR counter area
12	C	P24248E	End of 4248 PUB2

Figure 32. 4248 Printer PUB2 Extension

Bytes Dec	Bytes Hex	Label	Description
6	6	P23800X	Start of PUB2 extension
6	6	PB2SDR1	Channel data checks
7	7	PB2SDR2	Cont forms stacker misfolds
8	8	PB2SDR3	Burster/trimmer jams
9	9	PB2SDR4	No burst check
10	A	PB2SDR5	Burster/stacker jams
11	B	PB2SDRE	End of counters area
11	B	PB2DFLG	Default flags
		PB2DBRST	X'80' Default spec.=burst
12 — 15	C — F		Reserved
16 — 19	10 — 13	PB2DFCB	Default fcb id
20 — 23	14 — 17	PB2DCHAR	Default char. arrangement table id
24 — 27	18 — 1B	PB2DMDFY	Default copy modific. id
28 — 31	1C — 1F	PB2DFLSH	Default forms overlay frame id
32 — 35	20 — 23	PB2DFORM	Default paper forms id
36	24	PB2DFTE	End of default area
36 — 39	24 — 27	PB2WCGMS	Character sets presently load
40	28	PB2WMOD	WCGM# with modified character sets
		PB2WMOD0	X'80' WCGM0 contains a modified character set
		PB2WMOD1	40 WCGM1 cont. a mod.chr set
		PB2WMOD2	20 WCGM2 cont. a mod.chr set
		PB2WMOD3	10 WCGM3 cont. a mod.chr set
41	29	PB2FLAG1	First byte of flags
		PB2BURY	X'30' Burst = Y last specified
		PB2BURN	10 Burst = N last specified
		PB2UDCHK	08 DCHK=U was specified
42	2A	PB2FLAG2	Second byte of flags
		PB2TRCN	X'10' TRC=N was specified
		PB2TRCY	30 TRC=Y was specified
		PB2DEBTR	0E Debug = trac last specified
		PB2DEBDU	0A Debug = dump last specified
		PB2DEBTE	06 Debug = term last specified
		PB2DEBNO	02 Debug = none last specified
43	2B		Reserved
44 † 47	2C † 2F	PB2FCB	Currently loaded FCB id
48 † 63	30 † 3F	PB2CHAR	Character arrangement tables (CAT)
48 † 51	30 † 33	PB2CHAR1	Id of 1st CAT currently loaded
52 † 55	34 † 37	PB2CHAR2	Id of 2nd CAT currently loaded
56 † 59	38 † 3B	PB2CHAR3	Id of 3rd CAT currently loaded
60 † 63	3C † 3F	PB2CHAR4	Id of 4th CAT currently loaded
64 † 67	40 † 43	PB2CMCHR	Id of CAT used when loading current copymod
68 † 71	44 † 47	PB2CPMOD	Id of copymod currently loaded into the printer
72 † 75	48 † 4B	PB2FORMS	Id of paper form currently loaded
76 † 79	4C † 4F	PB2FLASH	Id of current forms overlay frame
80 † 87	50 † 57	PB2COPYG	Eight copy group count last received by setprint
88	58	PB2CINDX	Copy group id last received by setprint
89	59	PB2FLSHC	Flash count last received by setprint
90 † 91	5A † 5B		Reserved
92	5C	P23800E	End of 3800 PUB2

Figure 33. 3800 Printer PUB2 Extension

Bytes Dec	Bytes Hex	Label	Description
6 – 25	6 – 19	SDR3886	SDR counter area
26	1A	P23886E	End of 3886 PUB2

Figure 34. 3886 Optical Character Reader PUB2 Extension

Bytes Dec	Bytes Hex	Label	Description
6	6	P23890X	Start of PUB2 extension
6 – 15	6 – F	SDR3890	SDR counter area
16	10	P23890E	End of 3890 PUB2

Figure 35. 3890 Document Reader Extension

Bytes Dec	Bytes Hex	Label	Description
6	6	P2DISKX	Start of PUB2 extension
6	6	P2DFLG	Disk flags
		P2SDERRQ	X'80' Soft DASD error is queued
		P2DLOG	40 ERP requests error logged
7 – 8	7 – 8		Reserved
9	9	P2DMOD	Physical module identifier
10 – 15	A – F	P2DVOL	Volume serial number
16	10	P23330E	End of 3330 PUB2
16	10	P23340E	End of 3340 PUB2
16	10	P23350E	End of 3350 PUB2
16	10	P2FBAE	End of FBA PUB2
16	10	P2DISKE	End of 23xx PUB2

Figure 36. Disk Device PUB2 Extension

Bytes Dec	Bytes Hex	Label	Description
6–7	6–7	P2TNAME	Last 2 characters of name of A–transient that should run first for error condition processing. For some devices, defined only if P2TERP is on.
8	8	P2TFLG1	Flags for communication between supvr and A–transients during processing of an error condition. Bits defined below.
		P2TUNSOL	Bit 1 Error cond is unsolicited Not ready–to–ready interrupt. Set by supvr
		P2TERP	Bit 2 ERP is controlling device. Set by transients.
		P2TREPO	Bit 3 Retry I/O should be reposition Set by transients.
		P2TIEORG	Bit 4 Use original tie byte; if off the opposite tie is used. Set by transients
		P2TECPT	Bit 5 Intercept next I/O request. Set by transients
		P2TROR	Bit 6 Retry I/O should be read opposite. Set by transients
		P2TREST	Bit 7 Retry I/O should be restart of user's channel program. Set by transients.
9	9		Reserved
10–15	A– F	P2TVOL	Tape serial number
16–17	10–11	P2TBLK	Block length
18	12	P2TEMPR	Temporary read count
19	13	P2TEMPW	Temporary write count
20–21	14–15	P2ERG	Erase gap count
22–23	16–17	P2CLEAN	Cleaner action counts
24	18	P2NOISE	Noise record count
25	19	P2PRD	Permanent read errors
26	1A	P2PWT	Permanent write errors
27	1B	P23480E	End of 3480 PUB2 entry
27–29	1B–1D		Reserved
30	1E	P2ORGTIE	TIE original direction
31	1F	P2OPPTIE	TIE opposite direction
32–39	20–27	P2TWORKA	ERP Reposition CCW
40–67	28–43	P2TERP1	ERP Work area
68–87	44–57	P2SDR	SDR Counter area.
	44	SDR2400	24xx SDR counter area
	44	SDR3420	3410/20/22/30 SDR counter area
	58	P22400E	End of 24xx PUB2 Entry
	58	P23420E	End of 3420,etc. PUB2 Entry

Figure 37. Tape Device PUB2 Extension

Sense Byte 0

Sense byte 0 is shown in Figure 10 as an example of sense information. The complete contents of up to 32 sense bytes for a device is shown in the respective device manuals.

```
Sense Byte 0:

Bit 0: Command reject
Bit 1: Intervention required
Bit 2: Busout check except for 8809 tape, 3203 and 5203 printers, or
       2560 card machine
Bit 3: Equipment check except for MCR/OCR devices
Bit 4: Data check except for 1443 printer (type bar), 3210 and 3215
       operator consoles, or the 3881 or 3886 optical readers.
Bit 5: Overrun condition for disks and tapes, for 1442, 2501, 2520,
       or 2596 card devices, for MCR/OCR devices; type bar for 1443
       printer; buffer parity check for all printers except 3800;
       feed or machine check for 2560 card machine.
Bit 6: Track condition check for CKD disks; word count zero for tapes;
       load check for PRT1 and 3800 printers; non-recovery for
       1287,1288 OCRs; not initialized for 3886 OCR;
       unusual command sequence for 2540 card device and
       3881 OMR; auto select for 1419,1255,1259 MCRs and 1270 and
       1275 OCRs; no card available for 2560, 5424, and 5425 card
       machines; abnormal format reset for 3505 and 3525 card devices.
Bit 7: Channel 9 overflow for all printers; seek check for CKD disks;
       data converter check for 3410T7; permanent error for 3504,
       3505, and 3525; keyboard correction for 1287 and 1288 OCRs;
       RCP error for 3886 OCR; SIO batch numbering switch off for
       1419, 1255, and 1259 MCRs and for 1270 and 1275 OCRs; TIO
       document spacing error for 1287 OCR.
```

Figure 38. Sense Byte 0

Recorder File Record Formats (See List of Figures)

The following figures show in detail the various record formats on the recorder file IJSYSRC.

Displm. Dec	Displm. Hex	Length	Description
0	0	2	Header record identifier — hex FFFF
2	2	4	Low extent address of recorder file (CCHH)
6	6	4	High extent address of recorder file (CCHH)
10	A	1	Tracks per cylinder of the SYSREC device
11	B	7	Address of the record entry area (00CCHHR)
18	12	2	Remaining bytes on track just written on
20	14	2	Total number of bytes to be written on track
22	16	7	Address of the last record written (00CCHHR)
33	21	1	Device code of SYSREC hex 0B = 3350 hex 0C = 3375 hex 0E = 3380
34	22	4	Address of track where the 90 % full point is
38	26	1	Switches: Bit 0 Message warns that the 90 % full point is passed. Cleared by EREP when IJSYSRC cleared and header record rewritten. Bit 1 An emergency recording occurred due to system termination when file is full. Bit 2 Both channel and machine check records are recorded.
39	27	1	Precoded hex FF to check for valid header record

Figure 39. Header Record for IJSYSRC on CKD Device

Displm. Dec	Displm. Hex	Length	Description
0	0	2	Header record identifier — hex FFFF
2	2	4	Low extent of recorder file
6	6	4	High extent of recorder file
11	B	4	Address of the recording area of recorder file
22	16	4	Address of last record written in rec. area
33	21	1	Device code for FBA device — hex 0F
34	22	4	Block where the 90 % full point is
38	26	1	Switches: Bit 0 Message warns that the 90 % full point is passed. Cleared by EREP when IJSYSRC cleared and header record rewritten. Bit 1 Emergency recording for system end when the file is full. Bit 2 Both channel and machine check records are recorded.
39	27	1	Precoded hex FF to check for valid header record

Figure 40. Header Record for IJSYSRC on FBA Device

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Record type: x'30'
	1	1	1	Release level (see RFTABLE byte 7)
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
H	3	3	1	Record dependent switch – x'00' (unit check) Bit 1=0: Permanent error 1: Temporary error
E				
A				
D	6	6	1	'k of n': x'11' (1st of 1 total)
E	8	8	4	Date (x'00yydddF')
R	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID or zeros if: No user CCB available in the error queue PUB channel queue pointer is hex FF CCW address invalid in case of EXCP REAL
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data: x'03'
	49	31	3	Failing device address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes: 6
	64	40	6	Volume serial number
	70	46	2	Block length
	76	4C	1	Temporary reads count
	77	4D	1	Temporary writes count
	78	4E	2	SIO count for volume
	80	50	1	Permanent reads count
	81	51	1	Permanent writes count
	82	52	1	Noise blocks count
	83	53	1	Mode set count
	84	54	2	Erase gap count
	86	55	2	Cleaner action count
	88	58	10	Statistics counter area
	98	62	6	Sense byte data

Figure 41. Record for 2400T9 Tape on IJSYSRC (Unit Check)

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Record type: x'30'
	1	1	1	Release level (see RFTABLE byte 7)
	2	2	1	Record independent switches Bit 0=0: No more records to follow 4=0: Time field is in timer units
	3	3	1	Record dependent switches x'80': Device EOD record x'40': Counter overflow record x'04': Volume dismount record x'02': SVC-requested record
	6	6	1	'k of n': x'11' (1st of 1 total)
	8	8	4	Date (x'00yydddF')
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID (Counter overflow only, else zeros)
	32	20	8	Failing CCW (Counter overflow only, else zeros)
	40	28	8	CSW (Counter overflow only, else zeros)
	48	30	1	No. of doublewords of dev.dep. data: 3
	49	31	3	Failing device address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area: 10
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes: 6
	64	40	6	Volume serial number
	70	46	2	Block length
	76	4C	1	Temporary reads count
	77	4D	1	Temporary writes count
	78	4E	2	SIO count for volume
	80	50	1	Permanent reads count
	81	51	1	Permanent writes count
	82	52	1	Noise blocks count
	83	53	1	Mode set count
	84	54	2	Erase gaps count
	86	56	2	Cleaner actions count
	88	58	10	Statistics counter area
	98	62	6	Sense byte data

Figure 42. OBR for 2400T9 Tape (not Unit Check)

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Record type: x'30'
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
H				
E	3	3	1	Record dependent switch — x'00' (unit check)
A	6	6	1	'k of n': x'11' (1st of 1 total)
D	8	8	4	Date (x'00yydddF')
E	12	C	4	Timer units
R	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID or zeros if: No user CCB available in the error queue PUB channel queue pointer is hex FF CCW address invalid in case of EXCP REAL
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing device address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area: 20
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes: 9
	64	40	6	Volume serial number
	70	46	2	Block length
	76	4C	1	Temporary reads count
	77	4D	1	Temporary writes count
	78	4E	2	Start I/O count for volume
	80	50	1	Permanent reads count
	81	51	1	Permanent writes count
	82	52	1	Noise blocks count
	83	53	1	Mode set count
	84	54	2	Erase gaps count
	86	56	2	Cleaner actions count
	88	58	10	Statistics counter area. Figure 45 on page 119
	108	6C	9	Sense byte data

Figure 43. OBR for 3410/11, 3422, 3430 Tapes on IJSYSRC (Unit Check)

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Record type: x'30'
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dependent switches x'80': Device EOD record x'40': Counter overflow record x'04': Volume dismount record x'02': SVC–requested record
	6	6	1	'k of n': x'11' (1st of 1 total)
	8	8	4	Date (x'00yydddF')
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID (Counter overflow, else zero)
	32	20	8	Failing CCW (Counter overflow, else zero)
	40	28	8	CSW (Counter overflow, else zero)
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing device address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area: 20
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes: 9
	64	40	6	Volume serial number
	70	46	2	Block length
	76	4C	2	Temporary reads and writes count
	78	4E	2	Start I/O count for volume
	80	50	2	Permanent reads and writes count
	82	52	1	Noise blocks count
	83	53	1	Mode set count
	84	54	2	Erase gaps count
	86	55	2	Cleaner actions count
	88	58	20	Statistics counter area. Figure 45 on page 119
	108	6C	9	Sense byte data (counter overflow, else zeros)

Figure 44. OBR for 3410, etc. Tapes (not Unit Check)

Displ.into Counter	Area	Bits Used	From Sense Byte,Bit(s)	Type of Count
0	0	none	———	Spare byte
1	1	0–7	1,0	Noise
2	2	0–7	3,0	Read write VRC
3	3	0–7	3,1	MTE/LRCR
4	4	0–7	3,3	EDC/CRC
5	5	0–7	3,4	ENV CHK/VRC
6	6	0–3	0,5	Overrun
		4–7	3,2	Skew
		0–3	3,7	C–Compare
7	7	4_7	———	Spare
8	8	0		ID burst check
		1–6		Mask bit expansion
		7		Parity
9	9	0–7	2,0–7	Track in error mask bits
10	A	0–3	5,2	Write tape mark check
		4–7	5,4	Parity compare
11	B	0–3	5,5	Tachometer check
		4–7	5,6	False end mark
12	C	0_3	———	Spare
		4–7	8,1	Feed through check
13	D	0_3	———	Spare
		4–7	8,3	End velocity check
14	E	0–3	8,4	Read back data not detected
		4–7	8,5	Start velocity check
15	F	0_3	———	Spare
		4–7	8,7	Velocity retry
16	10	0_7	———	Spare byte
17	11	0_7	———	Spare byte
18	12	0–7	3,6	Backward
19	13	0–3	0,2	Bus out check
		4–7	4,0	Tape unit positioning check

Figure 45. SDR counter area - 3410/11, 3422, 3430 Tapes

	Displm. Dec	Displm. Hex	Length	Description
HEER	0	0	1	Record type: x'30'
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dep. switches: x'00' (unit check)
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID or zeros if: No user CCB available in the error queue PUB channel queue pointer is hex FF CCW address invalid in case of EXCP REAL
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing device address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area: 20
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes: 24
	64	40	6	Volume serial number
	70	46	2	Block length
	76	4C	1	Temporary reads count
	77	4D	1	Temporary writes count
	78	4E	2	Start I/O count for volume
	80	50	1	Permanent reads count
	81	51	1	Permanent writes count
	82	52	1	Noise blocks count
	83	53	1	Mode set count
	84	54	2	Erase gaps count
	86	56	2	Cleaner actions count
	88	58	20	Statistics counter area. Figure 48 on page 122
	108	6C	24	Sense byte data

Figure 46. OBR for 3420 Tape (Unit Check)

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Record Type: x'30'
	1	1	1	Release level (see RFTABLE displ.7)
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dep. switches x'80': Device EOD record x'40': Counter overflow record x'04': Volume dismount record x'02': SVC–requested record
	6	6	1	'k of n': x'11' (1st of 1 total)
	8	8	4	Date (x'00yydddF')
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID (Counter overflow, else zeros)
	32	20	8	Failing CCW (Counter overflow, else zeros)
	40	28	8	CSW (Counter overflow, else zeros)
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing device address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area: x'20'
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes: 24
	64	40	6	Volume serial number
	70	46	2	Block length
	76	4C	1	Temporary reads count
	77	4D	1	Temporary writes count
	78	4E	2	Start I/O count for volume
	80	50	1	Permanent reads count
	81	51	1	Permanent writes count
	82	52	1	Noise blocks count
	83	53	1	Mode set count
	84	54	2	Erase gaps count
	86	56	2	Cleaner actions count
	88	58	20	Statistics counter area. Figure 48 on page 122
	108	6C	24	Sense byte data (Counter overflow, else zeros)

Figure 47. OBR for 3420 Tape on IJSYSRC (not Unit Check)

Bytes 88–107 of Error Record:

Displ.into Counter	Area	Bits Used	From Sense Byte, Bit(s)	Type of Count
0	0	none	———	Spare byte
1	1	0–7	1,0	Noise
2	2	0–7	3,0	Read write VRC
3	3	0–7	3,1	MTE/LRCR
4	4	0–7	3,3	EDC/CRC
5	5	0–7	3,4	ENV CHK/VRC
6	6	0–3	0,5	Overrun
		4–7	3,2	Skew
7	7	0–3	3,7	C–Compare
		4–7	4,3	Write trigger VRC
8	8	0 0	5,3	PE ID burst check
		1–6		Mask bit expansion
		7		Parity
9	9	0–7	2,0–7	Track in error mask bits
10	A	0–3	5,2	Write tape mark check
		4–7	5,4	Start read check
11	B	0–3	5,5	Partial record
		4–7	5,6	Excessive postamble
12	C	0–3	8,0	IBG drop while writing
		4–7	8,1	Feed through check
13	D	0–3		Spare
		4–7	8,3	Early begin read back check
14	E	0–3	8,4	Early end read back check
		4–7	8,5	Slow begin read back check
15	F	0–3	8,6	Slow end read back check
		4–7	8,7	Velocity retry
16	10	0–3		Spare
		4–7	9,1	Velocity change during write
17	11	none	———	Spare byte
18	12	0–7	3,6	Backward
19	13	0–3	0,2	Bus cut check
		4–7	4,0	ALU hardware error

Figure 48. SDR Area for 3420 Tape OBR

	Displm. Dec	Displm. Hex	Length	Description
H E A D E R	0	0	1	Record Type: x'30'
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dep. switches: 0
	6	6	1	'k of n': x'11' (1st record of 1 total)
	8	8	4	Date — x'00yydddF'
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Jobname
	32	20	8	Failing CCW — 0 if cannot compute
	40	28	8	CSW
	48	30	1	Size of device dependent data: 1
	49	31	3	Secondary device address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of SDR counter area: 0
	57	39	3	Primary device address (cuu)
	60	3C	2	Number of I/O retries: 0
	62	3E	2	Number of sense bytes — 32
	64	40	6	Volume serial number
	70	46	2	Block length
	72	48	32	Sense bytes

Figure 49. OBR for 8809 or 9347 Tape

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Record type: x'90' (EOD/EOT) x'91' (counter overflow)
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dep. switches x'04' — Volume dismount record x'00' — counter overflow / EOD record
	4	4	1	MDR device type code: x'40' (8809) x'1F' (9347) Subid information: x'00'
	6	6	1	'k of n': x'11' (1st record of 1 total)
	8	8	4	Date — x'00yydddF'
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing device address (cuu)
	26	1A	6	Volume serial number
	32	20	2	Block length
	34	22	32	Sense bytes or buffered log data

Figure 50. MDR for 8809 or 9347 Tape

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Record type: x'30'
	1	1	1	Release level
	2	2	1	Record independent switches
				Bit 0=0: No more records to follow
H				Bit 4=0: Time field is in timer units
E	3	3	1	Record dep. switches: x'00'
A	6	6	1	'k of n': x'11' (1st record of 1 total)
D	8	8	4	Date — x'00yydddF'
E	12	C	4	Timer units
R	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Jobname
	32	20	8	Failing CCW; 0 if cannot compute
	40	28	8	CSW
	48	30	1	Size of device dependent data (dwords): 2
	49	31	3	Secondary device address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of SDR counter area: 0
	57	39	3	Primary device address (cuu)
	60	3C	2	Number of I/O retries: 0
	62	3E	2	Number of sense bytes: 32
	64	40	6	Volume serial number
	70	46	2	Block length: 0 (unavailable)
	72	48	8	Misc. device dependent data: 0
	80	50	32	Sense bytes

Figure 51. OBR for 3424 or 3480 Tape

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Record type: x'90' (EOD/EOT) x'91' (counter overflow)
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dep. switches: 0
	4	4	1	MDR device type code: x'44' (3424) x'42' (3480)
	5	5	1	Subid information: x'8C'
	6	6	1	'k of n': x'11' (1st record of 1 total)
	8	8	4	Date — x'00yydddF'
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing device address (cuu)
	26	1A	6	Volume serial number
	32	20	4	0
	36	24	2	Block length
	38	26	32	Log data

Figure 52. MDR for 3424 or 3480 Tape

	Displm. Dec	Displm. Hex	Length	Description
H E A D E R	0	0	1	Type 2 I/O device record key — hex 34
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dependent switch — hex 00 = unit check
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	2	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data — hex 02
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0002
	64	40	2	Start I/O count
	66	42	1	Temporary counter
	67	43	1	Mask byte
	68	44	1	TP code (failing)
	69	45	1	TP code (original)
	70	46	1	Flags
	71	47	1	2740–2
	72	48	8	Terminal name
	80	50	1	Failing sense/status byte 1 for 3270 remote
	81	51	1	Original sense/status byte 0 for 3270 remote
	82	52	4	SIO count since last record

Figure 53. Record for 1050,2701,02,03,2848,3277,7770 on IJSYSRC (Unit Check)

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Type 2 I/O device record key — hex 34
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dependen switches hex 80: Device EOD record hex 40: Counter overflow record hex 02: SVC—requested record
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID (8 bytes of hex 00)
	32	20	8	Failing CCW (8 bytes of hex 00)
	40	28	8	CSW (8 bytes of hex 00)
	48	30	1	No. of doublewords of dev.dep. data — hex 02
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0002
	64	40	2	SIO count
	66	42	1	Temporary counter
	67	43	1	Mask field — hex 00
	68	44	1	Failing TP code
	69	45	1	Original TP code
	70	46	1	Flags
	71	47	1	2740—2
	72	48	8	Terminal name (8 bytes of hex 00)
	80	50	1	Failing sense field
	81	51	1	Original sense field
	82	52	4	SIO count since last record

Figure 54. Record for 1050,2701,02,03,2848,3277,7770 on IJSYSRC (Other Error Conditions)

Format for Unit Check, Counter Overflow, EOD, and SVC Error Records

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dependent switches hex 00 — Unit check condition hex A0 — Device EOD hex 60 — Counter overflow hex 02 — SVC–requested
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date (00yydddF)
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
Second Part of Unit Check Record				
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of device dependent data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area — hex 0A
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0002
	64	40	10	Statistics counter area. Figure 56 on page 130
	74	4A	2	Sense byte data
	76	4C	4	SIO count since last record
Second Part of Other Type 1 I/O Device Records				
	24	18	2	VSE device type (PUB device code)
	26	1A	2	OS device class and type
	28	1C	1	Length of statistics counter area — hex 0A
	29	1D	3	Failing channel unit address (cuu)
	32	20	10	Statistics counter area. Figure 56 on page 130
	42	2A	4	SIO count since last record

Figure 55. Record for 1287, 1288, and 1419S on IJSYSRC (Various Conditions)

Displ.into Counter	Area	Bits Used	Type of Count
0	0	0–3	Temporary reads
		4–7	Temporary writes
1	1	0–3	Intervention required
		4–7	BUS out check
2	2	0–3	Equipment check
		4–7	Overrun
3	3	0–3	Non recovery (Auto select on 1419)
		4–7	
4	4	0–3	(End 1419)
		4–7	Late stacker select
5	5	0–3	No doc found
		4–7	
6	6	0–3	Invalid operation
		4–7	
7	7	0–3	
		4–7	Channel data check

Figure 56. Statistics Counter Area for 1287, 1288 and 1419S

Format for Unit Check, Counter Overflow, EOD, and SVC Error Records

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dependent switches hex 00 — Unit check record hex A0 — Device EOD record hex 60 — Counter overflow record hex 02 — SVC–requested record
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
Unit Check Record:				
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of device dependent data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	hex 0A — Bytes of statistics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0001
	64	40	10	Statistics counter area. Figure 58 on page 132
	74	4A	1	Sense byte data
	75	4B	4	SIO count since last record
Record for Other I/O Device Records:				
	24	18	2	VSE device type (PUB device code)
	26	1A	2	OS device class and type
	28	1C	1	Length of statistics counter area — hex 0A
	29	1D	3	Failing channel unit address (cuu)
	32	20	10	Statistics counter area. Figure 58 on page 132
	42	2A	4	SIO count since last record

Figure 57. Record for 1403,19,42,43,2501,20,40,96,3210,15, and 3881 on IJSYSRC (Various Conditions)

Displ.into Counter	Area	Bits Used	Type of Count
0	0	0–3	Temporary reads
		4–7	Temporary writes
1	1	0–3	Intervention required
		4–7	BUS out check
2	2	0–3	Equipment check
		4–7	Overrun
3	3	0–3	Non recovery
		4–7	
4	4	0–3	
		4–7	
5	5	0–3	
		4–7	
6	6	0–3	
		4–7	
7	7	0–3	
		4–7	Channel data check

Figure 58. Statistics Counter Area for 1403,19,42,43,2501,20,40,96,3210,15 and 3881

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer unit
	3	3	1	Record dependent switches
H				hex 00 — Unit check record
D				hex 60 — Counter overflow record
E				hex A0 — Device EOD record
D				hex 04 — Volume dismount record
E				hex 02 — SVC requested record
R	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
For Unit Check Record:				
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data — hex 03
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area — hex 0A
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries plus one
	62	3E	2	Number of sense bytes — hex 06
	64	40	6	Volume serial number
	72	48	6	Last seek address
	80	50	6	Actual home read address
	88	58	10	Statistics counter area. Figure 60 on page 134
	98	62	6	Sense byte data
	104	68	4	SIO count since last record

Figure 59 (Part 1 of 2). Record for 2311 and 2314 on IJSYSRC (Various Conditions)

For Other Records:				
	24	18	2	VSE device type (PUB device code)
	26	1A	2	OS device class and type
	28	1C	1	Length of statistics counter area — hex 0A
	29	1D	3	Failing channel unit address (cuu)
	32	20	10	Statistics counter area. Figure 60 on page 134
	42	2A	4	SIO count since last record

Figure 59 (Part 2 of 2). Record for 2311 and 2314 on IJSYSRC (Various Conditions)

Displ.into Counter	Area	Bits Used	Type of Count
0	0	0–3	Temporary reads
		4–7	Temporary writes
1	1	0–3	Intervention required
		4–7	BUS out check
2	2	0–3	Equipment check
		4–7	Overrun
3	3	0–3	Track condense
		4–7	Seek check
4	4	0–3	Unsafe
		4–7	
5	5	0–3	Serial data set erase
		4–7	Channel tag line
6	6	0–3	ALU
		4–7	Track overflow
7	7	0–3	Missing address marker
		4–7	Channel data check

Figure 60. Statistics Counter Area for 2311 and 2314

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 1 I/O device record key – hex 30
	1	1	1	Release level
	2	2	1	Record independent switches
H				Bit 0=0: No more records to follow
E				Bit 4=0: Time field is in timer units
A	3	3	1	Record dependent switches – hex 00 = unit check
D	6	6	1	1st digit: nth record; 2nd digit: of total no.
E				here: hex 11 (means first record of one)
R	8	8	4	Date – hex 00yydddF
	12	C	4	Timer unit
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area – hex 0A
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes – hex 0018
	64	40	1	Correlation number
	72	48	10	Statistics counter area
	82	52	24	Sense byte data
	106	6A	4	SIO count since last record

Figure 61. Record for PRT1 Printers on IJSYSRC (Unit Check)

Format for Counter Overflow, EOD, and SVC Error Records:

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dependent switches hex 80 — Device EOD record hex 40 — Counter overflow record hex 02 — SVC—requested record
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID (Counter overflow only, else zeros)
	32	20	8	Failing CCW (Counter overflow, else zeros)
	40	28	8	CSW (Counter overflow only, else zeros)
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0018
	64	40	1	Not used by VSE
	72	48	10	Statistics counter area. Figure 63 on page 136
	82	52	24	Sense byte data (Counter overflow, else zeros)
	106	6A	4	SIO count since last record

Figure 62. Record for PRT1 Printers on IJSYSRC (Various Conditions)

Displ.into Counter	Area	Bits Used	Type of Count
0	0	0–3	
		4–7	
1	1	0–3	USCAR sync
		4–7	BUS out check
2	2	0–3	PSE synch
		4–7	P1 RT fail
3	3	0–3	SKEW
		4–7	Command reject
4	4	0–3	Service aid
		4–7	Command retried
5	5	0–3	Command suppressed
		4–7	
6	6	0–3	
		4–7	
7	7	0–3	
		4–7	

Figure 63. Statistics Counter Area for PRT1 Printers

First Record:

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 3 I/O device record key — hex 91
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=1: More records to follow Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
H	3	3	1	(Record dependent switches) Not used here
E	4	4	1	Record dependent switches hex 04 = 3211 hex 10 = All others
A				
D				
E	6	6	1	1st digit: nth record; 2nd digit: of total no. hex 13 means first record of three hex 23 means second record of three, etc.
R				
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing channel unit address (cuu)
	26	1A	1	Used by PRT1 printers (same as sense byte 23): hex 00 = 3203–4 or 3203–5 hex FF = 3211 hex 22 = 3262 hex 23 = 3263 hex 10 = 3289–4
	27	1B	1	Buffer type ID — hex 01 = UCSB
	28	1C	172	UCSB for first and second record
	28	1C	168	UCSB for third record

Figure 64. Record for PRT1 Printers on IJSYSRC (UCSB Off-Load)

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 3 I/O device record key — hex 91
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
H	3	3	1	(Record dependent switches) Not used here
E	4	4	1	Record dependent switches hex 04 = 3211 hex 10 = All others
A				
D				
E	6	6	1	1st digit: nth record; 2nd digit: of total no.
R				here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing channel unit address (cuu)
	26	1A	1	Used by 3203–4/5 and 3211 PRT1 printers hex FF = 3211 hex 00 = 3203–4 or 3203–5
	27	1B	1	Buffer type ID — hex 03 = PLB
	28	1C	150	Check read buffer
	178	112	10	First 10 PLB error positions, left justified

Figure 65. Record for PRT1 Printers on IJSYSRC (PLB/Check Read Buffer Off-Load)

	Displm. Dec	Displm. Hex	Length	Description
H E A D E R	0	0	1	Type 3 I/O device record key — hex 91
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=1: Another record to follow 0=0: Last record Bit 4=0: Time field is in timer units
	4	4	1	Record dependent switches hex 04 = 3211 hex 10 = All others
	6	6	1	1st digit: nth record; 2nd digit: of total no. hex 12 = first record of two hex 22 = second record of two
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing channel unit address (cuu)
	26	1A	1	Used by PRT1 printers (same as sense byte 23): hex 00 = 3203–4 or 3203–5 hex FF = 3211 hex 22 = 3262 hex 23 = 3263 hex 10 = 3289–4
	27	1B	1	Buffer type ID — hex 02 = FCB
	28	1C	1	Forms control address reg.(only 1st record)
	29	1D	171	FCB in first record
	28	1C	84	FCB in second record

Figure 66. Record for PRT1 Printers on IJSYSRC (FCB Off-Load and Cancel)

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	I/O device record key — hex 91
	1	1	1	Release level
	2	2	1	Device independent switches Bit 0=0: No more records to follow in this record byte.
	3	3	1	Record dependent switches — X'04' = 3211
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing channel unit address (cuu)
	26	1A	1	Used by PRT1 printers — hex FF = 3211
	27	1B	1	Buffer type ID — hex 00 = FCAR (3211 only)
	28	1C	1	Forms control address register

Figure 67. Record for PRT1 Printers on IJSYSRC (FCB Off-Load, No Cancel)

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	I/O device record key — hex 91
	1	1	1	Release level
	2	2	1	Device independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	4	4	1	Record dependent switches. X'10' = 3289–4
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing channel unit address (cuu)
	26	1A	1	Used by PRT1 printers — hex 10=3289–4
	27	1B	1	Buffer type ID — hex 04 = AHO
	28	1C	17	Bit string reflecting the 132 print positions

Figure 68. Record for PRT1 Printers on IJSYSRC (Any Hammer On Off-Load Condition)

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
H	3	3	1	Record dep.switches — hex 00 = unit check Bit 1=0: Permanent error 1: Temporary error
E A D E	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
R	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statitics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries plus one
	62	3E	2	Number of sense bytes — hex 0018
	64	40	6	Volume serial number
	72	48	8	Last seek address or block no.(FBA) in error
	80	50	6	Home address read
	88	58	24	Sense byte data
	112	70	4	SIO count since last record

Figure 69. Record for 3350,75,80 and FBA on IJSYSRC (Unit Check)

Format for Counter Overflow, Volume Dismount, EOD, and SVC Error Records:

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Type 3 I/O device record key — hex 91
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	4	4	1	Record dependent switches device code: hex 0B = 3350 hex 0C = 3375 hex 0E = 3380 hex 0F = FBA
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing channel unit address (cuu)
	26	1A	6	Volume serial number
	32	20	24	Sense Byte Buffer Off-Load

Figure 70. Record for 3350,75,80 and FBA on IJSYSRC (Various Conditions)

Format for Unit Check, Counter Overflow, EOD, and SVC Error Records:

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dep. switches hex 00 — Unit check record hex A0 — Device EOD record hex 60 — Counter overflow record hex 02 — SVC—requested record
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
Unit Check Record:				
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statitics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0004
	64	40	4	Sense byte data
	68	8	4	SIO count since last record
Other I/O Device Records:				
	24	18	4	PUB device type — PUB bytes 4 and 5
	28	1C	1	Length of statistics counter area.
	29	1D	3	Failing channel unit address (cuu)
	32	20	4	SIO count since last record

Figure 71. Record for 3505, 3525 Card Devices on IJSYSRC (Various Conditions)

Format for Unit Check, Counter Overflow, EOD, and SVC Error Records:

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dep. switches hex 00 — Unit check hex A0 — Device EOD hex 60 — Counter overflow hex 02 — SVC—requested
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
Unit Check Record:				
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area — hex 08
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0006
	64	40	8	Statistics counter area. See next figure.
	72	48	6	Sense byte data
	78	4E	4	SIO count since last record
Other I/O Device Records:				
	24	18	2	VSE device type (PUB device code)
	26	1A	2	OS device class and type
	28	1C	1	Length of statitics counter area — hex 08
	29	1D	3	Failing channel unit address (cuu)
	32	20	8	Statistics counter area. Figure 73 on page 145
	42	2A	4	SIO count since last record

Figure 72. Record for 3540 on IJSYSRC (Various Conditions)

Counter		Condition
Dec	Hex	
4	4	Bus out check
6	6	No record found (with ID CRC error)
7	7	No record found (without ID CRC error)
8	8	Fast or slow index
9	9	Address mask ID failure
10	A	Data AM incorrect
11	B	Data CRC error
12	C	Control unit overrun

16 four-bit counters are used (8 bytes).

Figure 73. Use of Statistics Counters by 3540

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	2	Record independent switches Bit 0=0 No more records to follow Bit 4=0 Time field is in timer units
	3	3	1	Record dep. switches
H				hex 80 — Device EOD record
E				hex 40 — Counter overflow record
A				hex 02 — SVC-requested record
D				hex 00 — Unit check record
E	4	4	1	Temporary busout parity error — hex 40
R	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID (Unit check; else zeros)
	32	20	8	Failing CCW (unit check + counter overflow only; else zeros)
	40	28	8	CSW (unit check; else zeros)
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0018
	64	40	1	Not used by VSE
	72	48	10	Statistics counter area
	82	52	24	Sense byte data (unit check, else zeros)
	106	6A	4	SIO count since last record

Figure 74. Record for 3800 on IJSYSRC (Various Conditions)

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 3 I/O device record key — hex 91
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=1: More records to follow Bit 4=0: Time field is in timer units
H				
E	4	4	1	Record dep. switches — hex 0C = 3800 printer
A	6	6	1	1st digit: nth record; 2nd digit: of total no.
D				here: hex 11 (means first record of one)
E	8	8	4	Date — hex 00yydddF
R	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	2	Failing channel unit address (cuu)
	26	19	1	Number of non—zero entries (Maximum 16)
	27	1A	1	3800 internal error log — hex 04
	28	1B	128	16 entries of sense bytes
	156	9C	2	BTS count (Count of pages — in hundreds — burst since error log last emptied)
	158	9E	2	CFS fold count (Count of forms —in hundreds folded since error log last emptied)
	160	100	2	Number of 100 feet of paper processed since the error log was first emptied
	162	102	2	Device serial number

Figure 75. MDR Record for 3800 on IJSYSRC (Internal Error Log)

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
	3	3	1	Record dependent switches
H				Unit check
E				hex A0 — Device EOD
A				hex 60 — Counter overflow
D				hex 22 — SVC—requested
E	6	6	1	1st digit: nth record; 2nd digit: of total no.
R				here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
Unit Check Record:				
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data — hex 02
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area — hex 14
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 0006
	64	40	1	Failing command
	65	41	3	Related data bytes
	80	50	20	Statistics counter area. Figure 77 on page 148
	100	64	6	Sense bytes
	106	6A	4	SIO count since last record
Other I/O Device Records:				
	24	18	2	VSE device type (PUB device code)
	26	1A	2	OS device class and type
	28	1C	1	Length of statistics counter area
	29	1D	3	Failing channel unit address (cuu)
	32	20	20	Statistics counter area. Figure 77 on page 148
	52	34	4	SIO count since last record

Figure 76. Record for 3886 on IJSYSRC (Various Conditions)

Event	Counter		Condition
Intervention Required	1	1	Separator jam switch
	2	2	Separator timeout
	3	3	Document too long
	4	4	Aligner section timeout
	5	5	Intermediate transport timeout
	6	6	Stacker select error
	7	7	Hopper not in position
	8	8	Serial no update check
	9	9	Stacker A jam
	10	A	Stacker B jam
	11	B	Intermediate transport overrun
	12	C	Step motor stop error
	13	D	Stacker A or B jam
	14	E	Stepping motor start error
	15	F	Stepping motor speed error
	16	10	Increment timeout
	17	11	Stacker sel jam
	18	12	Clutch fail to pick
Equipment Check	20	14	Traverse limit switch
	21	15	Traverse time overrun
	22	16	Traverse emitter error
	23	17	Traverse speed error
	24	18	Traverse scan pitch control error
Mark Check	27	1B	Line mark not complete
	28	1C	Serial number print check
	29	1D	Line mark print check
	30	1E	Line mark detect error
	31	1F	Line mark miscompare
Incomplete Scan	32	20	Character out of scan
	33	21	Field delimiter not found
	34	22	Character on field boundary
	35	23	Video overrun
	36	24	Character buffer overflow
	37	25	Noise check
Non Recovery	39	27	Read station TM check
	40	28	Line mark request error
40 four—bit counters are used (20 bytes).			

Figure 77. Use of Statistics Counters by 3886

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Type 1 I/O device record key — hex 30
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 4=0: Time field is in timer units
H	3	3	1	Record dep. switches hex 00 = unit check record
E A D	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
E	8	8	4	Date — hex 00yydddF
R	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID
	32	20	8	Failing CCW
	40	28	8	CSW
	48	30	1	No. of doublewords of dev.dep. data
	49	31	3	Failing channel unit address (cuu)
	52	34	2	VSE device type (PUB device code)
	54	36	2	OS device class and type
	56	38	1	Length of statistics counter area
	57	39	3	Physical channel unit address (cuu)
	60	3C	2	Number of I/O retries
	62	3E	2	Number of sense bytes — hex 000D
	64	40	13	Sense data
	77	4D	4	SIO count since last record
The 5424 is identified by hex 34 at byte 52 in the record. No records are written for the 5425.				

Figure 78. Record for 5424 on IJSYSRC (Unit Check)

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Record type hex 50
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0. No more records to follow Bit 4=0. Time field is in timer units
H	3	3	1	Record dep. switches No meaning in IPL record.
E A D	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
E R	8	8	4	Date — 00YYDDF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	1	Subsystem ID
	28	1C	2	Reason code
	30	1E	2	Channel map
	32	20	8	Channel type assignments
	40	28	4	Highest storage address

Figure 79. IPL Record

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Record type hex 70
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=1: Another record to follow
H E				Bit 4=0: Time field is in timer units
A D	4	4	2	Flags: Bit 0=1: Channel end found pending
E R				1=1: Device end found pending
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU ID
	20	14	2	CPU model number
	22	16	2	MCEL length
	24	18	8	Job name
	32	20	3	Actual channel and unit address in EBCDIC
	35	23	3	Primary channel and unit address
	38	26	6	Volume ID
	44	2C	2	VSE device type (PUB device code)
	46	2E	2	OS device class and type
	48	30	8	Seconds used by the MIH to check for pending interrupts (in EBCDIC)

Figure 80. Missing Interrupt Records

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Record type hex 80
	1	1	1	Release level
H	2	2	1	Record independent switches
E				Bit 0=0: No more records to follow
A				Bit 4=0: Time field is in timer units
D	6	6	1	1st digit: nth record; 2nd digit: of total no.
E				here: hex 11 (means first record of one)
R	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length

Figure 81. System End of Day (SEOD) Record

	Displm. Dec	Displm. Hex	Length	Description
	0	0	1	Record type hex 10
	1	1	1	Release level
	2	2	1	Record independent switches
				Bit 0=0: No more records to follow
H				Bit 0=1: Another record to follow
E				Bit 3=1: 370XA mode record
A				Bit 4=0: Time field in timer units
D	3	3	1	Record dep. switches
E				Bit 1=1: Record incomplete (data missing)
R				Bit 2=1: System termination
	6	6	1	1st digit: nth record; 2nd digit: of total no.
				hex 12 = first record of 2
				hex 22 = second record of 2
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU model number
	22	16	2	MCEL length
First Record:				
	24	18	8	Program ID
	32	20	8	Job ID
	40	28	8	Program status word (PSW).
	48	30	152	Machine check independent logout, part 1
Second Record:				
	24	18	104	Machine check independent logout, part 2
	128	80	(0–?)	Machine check extended logout (CPU dependent)
				— evtl. a third record is required

Figure 82. Machine Check Records

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	0	Record type hex 20
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=0: No more records to follow Bit 0=1: Another record to follow Bit 3=1: 370XA mode record Bit 4=0: Time field in timer units
	3	3	1	Record dep. switches Bit 0=1: Message required. Always on for channel check records. Bit 1=1 Record incomplete Bit 2=1 System termination Bit 6=1 Recovery in progress
	6	6	1	1st digit: nth record; 2nd digit: of total no. here: hex 11 (means first record of one)
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU ID
	22	16	2	MCEL length
	24	18	8	Job ID
	32	20	16	Active I/O units at failure. A series of two-byte channel-unit addresses (0cuu)
	48	30	8	Failing CCW
	56	38	8	CSW
	64	40	4	ECSW
	68	44	2	VSE device type (PUB device code)
	70	46	2	OS device class and type
	72	48	1	Channel ID
	73	49	3	Failing channel unit address (cuu)
	76	4C	1	VSE/EREP use: Bit 0=1: I/O units invalid
	77	4D	1	Reserved for VSE use
	78	4E	2	I/O extended logout length

Figure 83. Channel Check Records

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Record type hex 25
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=1: Another record to follow Bit 0=0: No more records follow Bit 3=1: 370XA mode record Bit 4=0: Time field in timer units
	3	3	3	Record dep. switches (not used)
	6	6	1	1st digit: nth record; 2nd digit: of total no. eg.hex 12 = first record of 2 hex 22 = second record of 2
	7	7	1	reserved
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU model number
	22	16	2	Max length of CPU dependent MCEL area
	24	18	8	CSECT name of module doing recording (ILVRAS04)
	32	20	1	CRW recording code
	33	21	1	Flag byte 1: Bit 0=1: Hardware—stored CRW
	34	22	1	Flag byte 2
	35	23	1	CRW origin code: Bit 7=1: CRW—pending machine check
	36	24	2	Processor address where CRW is retrieved
	38	26	2	Reserved
	40	28	4	Channel Report Word
	44	2C	2	Device number
	46	2E	36	Device— and system—dependent data

Figure 84. Channel Report Word (CRW) Record

	Displm. Dec	Displm. Hex	Length	Description
HEADER	0	0	1	Record type hex A2
	1	1	1	Release level
	2	2	1	Record independent switches Bit 0=1: Another record to follow Bit 0=0: No more records follow Bit 3=1: 370XA mode record Bit 4=0: Time field in timer units
	3	3	3	Record dep. switches (not used)
	6	6	1	1st digit: nth record; 2nd digit: of total no. eg.hex 12 = first record of 2 hex 22 = second record of 2
	7	7	1	reserved
	8	8	4	Date — hex 00yydddF
	12	C	4	Timer units
	16	10	1	CPU version identifier
	17	11	3	CPU serial number
	20	14	2	CPU model number
	22	16	2	Max length of CPU dependent MCEL area
	24	18	104	A2 Information

Figure 85. Link Maintenance Information (A2) Record

Diagnostics

Hard Wait Codes

Byte 0	Byte 1	Byte 2	Byte 3	Description
X'C1'	X'00'	A,I,S	not used	Irrecoverable machine check.
X'C2'	X'00'	A,I,S	not used	Irrecoverable channel check during FETCH.
X'C3'	X'00'	A,I,S	not used	Irrecoverable channel check on paging channel.
X'C5'	X'00'	A,I,S	not used	No ECSW stored.
X'C7'	X'00'	A,I,S	not used	Channel failure; channel address invalid.
X'C8'	X'00'	A,I,S	not used	Channel failure on SYSLOG

Notes:

A X'C1' - SYSREC recording unsuccessful (No record written)
I X'C9' - SYSREC recording incomplete (Not all records written)
S X'E2' - SYSREC recording successfully completed

Figure 86. MCH/CCH Wait Codes

Message-to-Phase Cross Reference

Table 7 on page 156 lists, for each message which error recovery and recording transients issue, all the transients that select it to be issued. Selecting a message ordinarily means putting a code that identifies it in a control block field. The transients listed in this table specifically select the listed message. A transient which simply moves a message code from one control block field to another does not appear in the table.

The messages in the table are not necessarily selected by error recovery and recording transients. They may also be selected by the supervisor.

Table 7 (Page 1 of 4). Message to phase cross-reference

Msg No	Description	Phases
0P08	Intervention Required	$$ABERAE $$ABERAJ $$ABERAN $$ABERBA $$ABERRF $$ABERRG $$ABERRS $$ABERRT $$ABERRY $$ABERR7
0P09	Busout Check	$$ABERAE $$ABERAN $$ABERBA $$ABERRF $$ABERRG $$ABERRS $$ABERRT $$ABERRY $$ABERR7
0P10	Equipment Check	$$ABERAC $$ABERAE $$ABERAF $$ABERAN $$ABERRF $$ABERRG $$ABERRY $$ABERR7
0P11	Data Check	$$ABERAB $$ABERAC $$ABERAF $$ABERAI $$ABERAJ $$ABERBA $$ABERRF $$ABERRG $$ABERRY $$ABERR7
0P14	Overrun	$$ABERAE $$ABERRT $$ABERRY
0P17	File Protection Violation	$$ABERAE
0P18	Command Reject	$$ABERAE $$ABERAN $$ABERBA $$ABERRF $$ABERRG $$ABERRS $$ABERRT $$ABERRY $$ABERR7

Table 7 (Page 2 of 4). Message to phase cross-reference

Msg No	Description	Phases
0P19	Undetermined Error	$$ABERAE $$ABERAJ $$ABERAN $$ABERBA $$ABERRF $$ABERRG $$ABERRS $$ABERRT $$ABERRY $$ABERR7
0P20	Error in Recovery Attempt	$$ABERAC $$ABERAF $$ABERAG $$ABERRF
0P24	Channel Program Check	$$ABERAE $$ABERRA
0P25	Channel Protection Check	$$ABERAE $$ABERRA
0P27	Unknown Device	$$ABERRA $$ABERRY
0P28	Channel Data Check	$$ABERAE $$ABERAN $$ABERBA $$ABERRF $$ABERRG $$ABERRS $$ABERRT $$ABERRY $$ABERR7
0P29	Backward Into Loadpoint	$$ABERAE
0P30	Tape Conversion Check	$$ABERAE
0P31	Device Not Operational	$$ABERAE $$ABERRA
0P32	Tape Cannot Be Read	$$ABERAE
0P33	Parity Error in Printer Buffer	$$ABERRF $$ABERRY
0P34	Batch Numbering Switch Off in MICR	$$ABERRS
0P35	Non Recovery	$$ABERRT $$ABERAN
0P37	Disengage Failure	$$ABERRS
0P38	Invalid Font	$$ABERRT
0P39	Bad Volume	$$ABERAC
0P41	Buffer Load Check	$$ABERBA $$ABERRF
0P42	DSE Failed	$$ABERAE $$ABERAG

Table 7 (Page 3 of 4). Message to phase cross-reference

Msg No	Description	Phases
0P43	Tape Volume Changed	$$ABERAG $$ABERAJ
0P44	ID Mark Check	$$ABERAC $$ABERAJ
0P46	Lost Positioning	$$ABERAC $$ABERAJ
0P47	Unexpected Intervention	$$ABERAE $$ABERAJ
0P48	Format Reset	$$ABERRG
0P49	Permanent Error	$$ABERRG $$ABERR7
0P50	No Channel Code Found	$$ABERRX
0P51	Mark Check	$$ABERAN
0P52	Invalid Format	$$ABERAN
0P53	Recognition Control Program Error	$$ABERAN
0P54	Not ICPL'd	$$ABERR7
0P58	Punch Data Check	$$ABERRY
0T00	Threshold on Recorder File Reached	$$ABERA1 $$ABERA2 $$ABERA3 $$ABERA9 $$RAST08 $$RAST11 $$RAST16
0T03	Error on Recorder File	$$ABERA1 $$ABERA2 $$ABERA3 $$ABERA9 $$RAST08 $$RAST11 $$RAST16
0T05	Recorder File Full	$$ABERA1 $$ABERA2 $$ABERA3 $$ABERA9 $$RAST08 $$RAST11 $$RAST16
0T08	C40 Buffer Pages Deleted	$$RAST03 $$RAST11
0T09	Successful Recovery from Machine Check	$$RAST03 $$RAST11

Table 7 (Page 4 of 4). Message to phase cross-reference

Msg No	Description	Phases
0T10	Channel Error Recovered	$$RAST02 $$RAST05 $$RAST06 $$RAST07 $$RAST10 $$RAST12
0T11	Hard Wait Code	$$RAST00 $$RAST01
0T12	Irrecoverable Channel Error	$$RAST02 $$RAST05 $$RAST06 $$RAST07 $$RAST10 $$RAST12
0T13	Channel Error on cuu	$$RAST05 $$RAST06 $$RAST10
0T14	Clock Damage. All Modes Quiet.	$$RAST03 $$RAST11
0T15	MCAR Repair Failed	$$RAST13
0T18	Timer Damaged	$$RAST03 $$RAST11
0T19	Allocation of x Has Become Invalid	$$RAST13
0T20	PFIX limit reduced	$$RAST13
0T21	System Performance Degradation	$$RAST03
0T22	Clock and/or timer damage	$$RAST03
0T24	System running on UPS	$$RAST03
0T25	Utility POWER restored	$$RAST03
0T30	Channel report word lost due to overflow condition	$$RAST04
0T31	CHPID xx Alert, unsolicited malfunction interrupt	$$RAST04
0T32	CHPID xx Alert, no associated subchannel for device	$$RAST04
0T33	CHPID xx Alert, channel path permanent error	$$RAST04
0T34	CHPID xx Alert, channel path terminal	$$RAST04
0T35	Event information lost due to overflow condition	$$RAST04
4E10	tape error statistics	$$ABERAA $$ABERA2

Index

Special Characters

Numerics

A

B

C

D

E

F

H

I

L

M

N

O

P

R

S

T

U

V

Communicating Your Comments to IBM

IBM VSE/Enterprise Systems Architecture
VSE Central Functions
Error Recovery and Recording Transients
Diagnosis Reference
Version 6 Release 1

Publication No. SC33-6326-00

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of the book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF form and either send it postage-paid in the United States, or directly to:

  IBM Deutschland Entwicklung GmbH
  Department 3248
  Schoenaicher Strasse 220
  D-71032 Boeblingen
  Federal Republic of Germany

- If you prefer to send comments by FAX, use this number:
  - (Germany): 07031-16-3456
  - (Other countries): (+49)+7031-16-3456
- If you prefer to send comments electronically, use this network ID:

  INTERNET: s390id@de.ibm.com

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Readers' Comments — We'd Like to Hear from You

IBM VSE/Enterprise Systems Architecture
VSE Central Functions
Error Recovery and Recording Transients
Diagnosis Reference
Version 6 Release 1

Publication No. SC33-6326-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	□	□	□	□	□

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	□	□	□	□	□
Complete	□	□	□	□	□
Easy to find	□	□	□	□	□
Easy to understand	□	□	□	□	□
Well organized	□	□	□	□	□
Applicable to your tasks	□	□	□	□	□

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? □ Yes □ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
SC33-6326-00

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

PLACE
POSTAGE
STAMP
HERE

IBM Deutschland Entwicklung GmbH
Department 3248
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

File Number: S/370 9370-37
Program Number: 5686-066

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC33-6326-00