

IBM Enterprise Storage Server™



# System/390 Command Reference 2105 Models E10, E20, F10, and F20



IBM Enterprise Storage Server™



# System/390 Command Reference 2105 Models E10, E20, F10, and F20

**Note!**

Before using this information and the product it supports, be sure to read "Safety and environmental notices" on page ix and the general information under "Notices" on page 223.

**Second Edition (June 2000)**

This edition replaces SC26-7298-00.

Publications are not stocked at the address given below. If you want additional IBM publications, ask your IBM representative or write to the IBM branch office serving your locality.

A form for your comments is provided at the back of this document. If the form has been removed, you may address comments to:

International Business Machines Corporation  
RCF Processing Department  
G26/050  
5600 Cottle Road  
San Jose, CA 95193-0001  
U.S.A.  
  
FAX: 1-800-426-6209

You can also send your comments electronically to:

starpubs@us.ibm.com

For additional information on IBM storage products, visit the Web site at:

<http://www.ibm.com/storage>

© Copyright International Business Machines Corporation 1999, 2000. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>Safety and environmental notices</b> . . . . .	ix
Safety notices . . . . .	ix
Product recycling . . . . .	ix
Product disposal . . . . .	ix
<b>About this book</b> . . . . .	xi
Who should use this book . . . . .	xi
Publications . . . . .	xi
The ESS product library . . . . .	xi
Ordering publications . . . . .	xii
Related publications . . . . .	xii
Web sites . . . . .	xiii
<b>Summary of Changes</b> . . . . .	xv
<b>Chapter 1. IBM Enterprise Storage Server overview</b> . . . . .	1
Host systems supported by the IBM ESS . . . . .	3
SCSI host systems . . . . .	3
Fibre-channel host systems . . . . .	3
S/390 host systems . . . . .	4
Web interfaces . . . . .	4
Web connection security . . . . .	4
IBM Enterprise Storage Server Network . . . . .	4
Accessing ESS Specialist and ESS Copy Services . . . . .	5
ESS Specialist . . . . .	6
ESS Copy Services . . . . .	6
ESS Expert . . . . .	7
Service interface . . . . .	7
Remote services support . . . . .	7
Configuration . . . . .	8
Introducing the IBM Subsystem Device Driver . . . . .	8
<b>Chapter 2. Elements of data access</b> . . . . .	11
Data groupings . . . . .	11
Record format . . . . .	11
Count area . . . . .	12
Key area . . . . .	12
Data area . . . . .	13
Track format . . . . .	13
Track address . . . . .	13
Functional track groups . . . . .	13
Index . . . . .	14
End-of-Track . . . . .	14
Home Address - HA . . . . .	14
Record Zero - R0 . . . . .	14
Data access . . . . .	14
Track accessing . . . . .	14
Orientation . . . . .	15
Record accessing . . . . .	16
Data transfer . . . . .	16

Define Extent . . . . .	16
Locate Record/Locate Record Extended . . . . .	16
Formatting . . . . .	17
Erase . . . . .	17
File protection . . . . .	17
End-Of-File . . . . .	18
<b>Chapter 3. Command descriptions . . . . .</b>	<b>19</b>
Command summary . . . . .	19
Exception conditions . . . . .	22
Command operations . . . . .	22
Multi-track operations . . . . .	22
Addressing and control commands . . . . .	23
Define Extent . . . . .	23
Locate Record . . . . .	29
Locate Record Extended . . . . .	41
Seek commands - general . . . . .	50
Seek . . . . .	51
Seek Cylinder . . . . .	51
Seek Head . . . . .	52
Set File Mask . . . . .	53
Set Sector . . . . .	53
Read Sector . . . . .	54
Read Device Characteristics . . . . .	54
Read commands . . . . .	60
Read Special Home Address . . . . .	61
Read Home Address . . . . .	62
Read Record Zero . . . . .	62
Read Count Key and Data . . . . .	63
Read Key and Data . . . . .	64
Read Data . . . . .	65
Read Count . . . . .	66
Read Multiple Count Key and Data . . . . .	67
Read Track . . . . .	68
Read IPL . . . . .	70
Search commands . . . . .	70
Search Home Address Equal . . . . .	71
Search ID Equal . . . . .	72
Search ID High . . . . .	73
Search ID Equal or High . . . . .	73
Search Key Equal . . . . .	73
Search Key High . . . . .	74
Search Key Equal or High . . . . .	74
Write commands . . . . .	74
Write Special Home Address . . . . .	75
Write Home Address . . . . .	76
Write Record Zero . . . . .	76
Write Count, Key, and Data . . . . .	77
Write Count, Key, and Data Next Track . . . . .	79
Erase . . . . .	80
Write Update Key and Data . . . . .	81
Write Key and Data . . . . .	82
Write Update Data . . . . .	83
Write Data . . . . .	84

Sense commands . . . . .	86
Sense . . . . .	86
Sense ID . . . . .	87
Miscellaneous commands . . . . .	88
No-Operation . . . . .	88
Read and Reset Buffered Log . . . . .	89
Diagnostic Control . . . . .	90
Read Configuration Data . . . . .	95
Path control commands . . . . .	99
Device Reserve . . . . .	99
Device Release . . . . .	100
Unconditional Reserve . . . . .	101
Reset Allegiance . . . . .	102
Set Path Group ID (SPID) . . . . .	104
Sense Path Group ID (SNID) . . . . .	105
Suspend Multi-path Reconnection . . . . .	106
Subsystem commands . . . . .	107
Set Subsystem Mode . . . . .	107
Perform Subsystem Function . . . . .	111
Sense Subsystem Status . . . . .	134
Read Subsystem Data . . . . .	139
Read Message ID . . . . .	153
<b>Chapter 4. Multi-Path Lock Facility . . . . .</b>	<b>155</b>
Multi-Path Lock structure . . . . .	155
Multi-Path Lock connect and disconnect . . . . .	155
Creating and deleting Multi-Path Locks . . . . .	156
Multi-Path Lock state transformations . . . . .	156
Multi-Path Lock responses . . . . .	159
Utility functions . . . . .	159
MPLF State transformation tables . . . . .	159
MPLF terminology . . . . .	165
<b>Chapter 5. Sense conditions . . . . .</b>	<b>169</b>
Sense data summary . . . . .	169
24 Byte Compatibility sense data summary . . . . .	170
ECKD 32 Byte sense data summary . . . . .	171
24 Byte Compatibility sense data . . . . .	172
24 Byte Compatibility Common sense bytes . . . . .	172
Format 0 - Program or System Check . . . . .	180
Format 2 - ESS Equipment Check . . . . .	183
Format 3 - ESS Control Check . . . . .	184
Format F - Subsystem Data . . . . .	185
ECKD 32 Byte format . . . . .	189
ECKD 32 Byte Common sense bytes . . . . .	189
Exception Class dependent data . . . . .	195
<b>Chapter 6. Error recovery procedure . . . . .</b>	<b>203</b>
Bytes 0-3, ERP description . . . . .	203
Error condition table . . . . .	203
Channel error conditions . . . . .	205
Error recovery actions . . . . .	205
Error message . . . . .	209
Coded ERP description . . . . .	212

Coded program actions . . . . .	213
<b>Appendix A. Device data, geometries, and facilities</b> . . . . .	<b>219</b>
<b>Notices</b> . . . . .	<b>223</b>
<b>Trademarks</b> . . . . .	<b>225</b>
Electronic emission notices . . . . .	225
Federal Communications Commission (FCC) statement . . . . .	225
Industry Canada compliance statement . . . . .	226
European Community compliance statement . . . . .	226
Germany only . . . . .	226
Japanese Voluntary Control Council for Interference (VCCI) Class . . . . .	227
Korean Government Ministry of Communication (MOC) statement . . . . .	227
Taiwan Class A Compliance statement . . . . .	227
<b>IBM Agreement for Licensed Internal Code</b> . . . . .	<b>229</b>
Actions You Must Not Take . . . . .	229
<b>Glossary</b> . . . . .	<b>231</b>
<b>Index</b> . . . . .	<b>237</b>



---

# Figures

1.	Models E10, E20, F10 and F20 front and rear views	2
2.	Expansion enclosure, front and rear views	2
3.	Multi-path connections between host server and ESS LUNs	9
4.	Access authorization for track groups	13
5.	Channel commands listed by function	19
6.	Numeric list of channel commands	21
7.	Define Extent, Byte 0, Mask Byte	24
8.	Define Extent, Byte 1, Global Attributes	25
9.	Define Extent, Byte 6, Global Attributes Additional	27
10.	Define Extent, Byte 7, Global Attributes Extended	28
11.	Locate Record, Byte 0, Operation Byte	31
12.	Locate Record Operation Byte Values	32
13.	Locate Record, Byte 1, Auxiliary Byte	32
14.	Locate Record Extended, Byte 0, Operation Byte	43
15.	Locate Record Extended Operation Byte Values	44
16.	Locate Record Extended, Byte 1, Auxiliary Byte	44
17.	Read and Reset Buffered Log	90
18.	Diagnostic Control subcommands and their chaining and parameter requirements	92
19.	Configuration data	95
20.	Control byte definition	105
21.	Path State byte definition	106
22.	Parameter bytes for prepare to Read Lock Data Order X'22'	120
23.	Parameter bytes for Set Lock State Order X'30'	123
24.	Parameter bytes for Purge Lock Order X'31'	125
25.	Parameter bytes for Unlock Order X'32'	127
26.	Parameter bytes for Connect Order X'33'	129
27.	Parameter bytes for Disconnect Order X'34'	132
28.	Sense Subsystem Status, Byte 0, Format of Sense Subsystem Status	135
29.	Sense Subsystem Status, Byte 4, Overall Caching Status	135
30.	Sense Subsystem Status, Byte 5, non-volatile storage Status	136
31.	Sense Subsystem Status, Byte 26, Device Status - Group 1, 1/2	137
32.	Sense Subsystem Status, Byte 27, Device Status - Group 1, 2/2	137
33.	Sense Subsystem Status, Byte 27, Device Status - Group 1, 2/2	138
34.	Sense Subsystem Status, Byte 36, Device Status - Group 2	138
35.	Sense Subsystem Status, Byte 37, Global Status	139
36.	Path Status	140
37.	Performance Statistics	141
38.	Delayed Response Format	143
39.	Multi-Path Lock Facility Message Format	144
40.	Multi-Path Lock Facility Message Reason Codes	146
41.	Logical Path Status Entry	148
42.	Read Multi-Path Lock User ID Data Format	149
43.	Read Multi-Path Lock Data Format	149
44.	Read Multi-Path Lock Data – Lock Entry Format	150
45.	Read MPLF Partition Data	150
46.	Node Descriptor Record	151
47.	Set Lock State example	157
48.	Promote/Demote example	158
49.	Unlock example	159

50.	Lock State Compatibility Matrix	160
51.	Sense information summary	170
52.	ECKD 32 Byte Exception Class	171
53.	ECKD 32 Byte sense information summary	171
54.	Coded message code generation	179
55.	Definition of Byte 3 for SIM-Format Sense Data	191
56.	Sense Byte 6, Content and Format	192
57.	Non-SIM message codes	195
58.	Error Condition Table	204
59.	Messages defined by Sense	210
60.	Definitions of abbreviations for messages defined by Sense	210
61.	Device Characteristics	220
62.	Sense ID Values	221
63.	Device Geometry and Facilities	221
64.	Device Track Address Ranges	222

---

## Safety and environmental notices

This section contains information about:

- Safety notices used in the ESS publications
- Environmental guidelines for this product

---

### Safety notices

For a translation of the danger, caution, and attention notices, see *IBM Storage Solution Safety Notices*, GC26-7229.

Translated notices are easy to locate in the safety notice translations as they are in numeric order. Look for the ID numbers (**1000** and **1001**) in the following examples.

#### **DANGER**

**A danger notice indicates the presence of a hazard that has the potential of causing death or serious personal injury.**

**1000**

#### **CAUTION:**

**A caution notice indicates the presence of a hazard that has the potential of causing moderate or minor personal injury.**

**1001**

#### **Attention:**

An attention notice indicates the possibility of damage to a program, device, system, or data.

**1002**

---

### Product recycling

This unit contains recyclable materials. These materials should be recycled where processing sites are available and according to local regulations. In some areas IBM provides a product take-back program that ensures proper handling of the product. Contact your IBM representative for more information.

---

### Product disposal

This unit may contain batteries. Remove and discard these batteries, or recycle them, according to local regulations.



---

## About this book

This book describes the functions of the IBM Enterprise Storage Server™ (ESS) and provides reference information for System/390® hosts, such as channel commands, sense bytes, and error recovery procedures.

A glossary at the back of this book defines the terms used in this book. This glossary includes terms for other Seascope™ solution products.

---

## Who should use this book

This publication documents command parameters and status information particular to the ESS. This information is useful to system programmers, system operators, and service support representatives who need to interpret error messages and codes. The messages will help determine if errors are caused by users, the operator, media, or hardware.

You can install and operate the ESS in supported environments without knowledge of the information in this publication.

---

## Publications

This section describes the ESS library and publications for related products. It also gives ordering information for these publications.

### The ESS product library

The ESS is an IBM Seascope architecture-based product. See the following publications for more information on the ESS:

- *IBM Enterprise Storage Server Introduction and Planning Guide, Models E10 and E20, GC26-7294*

This guide introduces the product and lists the features you can order. It also provides guidelines on planning for the installation and configuration of the ESS.

- *IBM Enterprise Storage Server User's Guide, Models E10 and E20, SC26-7295*

This guide provides instructions for setting up and operating the ESS.

- *IBM Enterprise Storage Server Host Systems Attachment Guide, Models E10 and E20, SC26-7296*

This guide provides guidelines for attaching the ESS to your host system.

- *IBM Enterprise Storage Server SCSI Command Reference, Models E10 and E20, SC26-7297*

This book describes the functions of the ESS. It provides reference information for UNIX™ and Application System/400® (AS/400®) hosts, such as channel commands, sense bytes, and error recovery procedures.

- *IBM Enterprise Storage Server System/390 Command Reference, Models E10 and E20, SC26-7298*

This book describes the functions of the ESS and provides reference information for System/390 hosts, such as channel commands, sense bytes, and error recovery procedures.

- *IBM Enterprise Storage Server Web Interface User's Guide*, SC26-7346

This book provides instructions for using the IBM StorWatch™ Enterprise Storage Server Web interface, ESS Specialist.

- *IBM Storage Solutions Safety Notices*, GC26-7229

This book provides translations of the danger notices and caution notices that IBM uses in ESS publications.

- *IBM Enterprise Storage Server Configuration Planner*, SC26-7353

This book provides work sheets for planning the logical configuration of the ESS. This book is not available in hard copy. This book is only available from the following Web site:

<http://www.ibm.com/storage/ess/ess.htm>

- *IBM Enterprise Storage Server Quick Configuration Guide*, SC26-7354

This booklet provides flow charts for using the StorWatch Enterprise Storage Server Specialist. The flow charts provide a high-level view of the tasks the IBM service support representative performs during initial logical configuration. You can also use the flow charts for tasks that you might perform during modification of the logical configuration.

The hard copy print of this booklet is a 9-inch X 4-inch fanfold.

## Ordering publications

All of the publications that are listed in the "The ESS product library" on page xi are available on a compact disc that comes with the ESS, unless otherwise noted. You can also order a hard copy of each publication. For publications on compact disc, order the *IBM Enterprise Storage Server Customer Documents compact disc*, SK2T-8770.

All the customer documents listed above are available on the following ESS Web site:

<http://www.ibm.com/storage/ess/ess.htm>

## Related publications

The following related publications are also available:

- *DFSMS/MVS V1R5 Software Support for the IBM Enterprise Storage Server*, SC26-7318

This book describes the changes you need to make to operating systems for System/390 hosts for support of the ESS functions.

- *IBM Subsystem Device Driver*

This book describes how to use the IBM Subsystem Device Driver on open host systems to enhance performance and availability on the ESS. The IBM subsystem device driver creates redundant paths for shared LUNs. The subsystem device driver permits applications to run without interruption when path errors occur. It balances the work load across paths. And, it integrates with applications transparently.

This publication is not available in hard copy. However, it is included on the compact disc that is delivered with your ESS. You can also obtain a copy from the following Web site:

<http://www.storage.ibm.com/hardsoft/products/ess/ess.htm>

- *IBM Storage Area Network Data Gateway Installation and User's Guide, 2108 Model G07, SC26-7304*

This book introduces the IBM Storage Area Network (SAN) Data Gateway and provides the procedures for installing and using it.

- *IBM Advanced Copy Services, SC35-0355*

This book provides detailed information on Copy Services functions such as peer-to-peer remote copy (PPRC), extended remote copy (XRC), and FlashCopy™.

- *IBM Enterprise Storage Server, SG24-5465*

This book, from the IBM International Technical Support Organization, describes the ESS, its components, and its functions.

- *The IBM Enterprise Storage Server in Your Environment, SG24-5420*

This book, from the IBM International Technical Support Organization, provides information about ESS performance in various environments.

- *IBM S/360, S/370, and S/390 Channel to Controller Original Equipment Manufacturer's Information, GA22-6974*

This publication provides a description of the interface between S/360, S/370™, and S/390™ channels and controllers that are designed to operate with this interface. This publication covers the parallel I/O interface. It does not cover the interface between the controller and the I/O device, nor does it cover the ESCON-I/O interface for the S/390.

This publication is for designers of programs and equipment that are associated with parallel I/O interfaces, and for service personnel that maintain the equipment. Anyone concerned with the functional details of this interface will also find it useful.

- *Fibre Transport Services (FTS) Direct Attach, Physical and Configuration Planning, GA22-7234*

This publication provides information on fibre optic and ESCON® trunking systems.

---

## Web sites

Additional information is available on the following Web sites:

- IBM Storage:

Visit the SUPPORT pages at the IBM Storage home page:

<http://www.storage.ibm.com/>

View and print IBM Enterprise Storage Server publications, and get specific details on models and operating systems that the ESS supports:

<http://www.ibm.com/storage/ess>





---

## Summary of Changes

Vertical revision bars in the left margin indicate technical changes to this document. Minor editorial changes are not marked with vertical revision bars.

### *June 2000*

The changes for this publication include the following:

- Changes have been made to Chapter 1, “IBM Enterprise Storage Server overview” on page 1 to include overview information on the Models F10 and F20.
- Minor revisions have also been made to Chapter 3 and to Chapter 5.



---

## Chapter 1. IBM Enterprise Storage Server overview

This section provides an overview of the IBM Enterprise Storage Server (ESS) and describes its interfaces and components.

The ESS is a member of the Seascape product family of storage servers. The ESS provides integrated caching and a choice of either 8 GB or 16 GB cache. It also provides redundant array of independent disks (RAID) support for the disk drive modules (DDMs). The DDMS are attached through a serial storage architecture (SSA) interface.

The ESS provides:

- RAID 5 or non-RAID
- Fast SSA drives
- Fast RISC processors
- Fault tolerant system
- Storage sharing S/390 and open systems
- OS/390® parallel I/O
- Instant copy solutions
- Disaster recovery solutions

The Model E20 with expansion enclosures provides up to 11 terabytes (TB)<sup>1</sup> of storage capacity, with a choice of 9.1, 18.2, or 36.4 gigabyte (GB)<sup>2</sup> DDMs. See Figure 1 on page 2 and Figure 2 on page 2 for illustrations of the ESS models and the expansion enclosure. When referring to DDM capacity, terabytes (TB) means 1 000 000 000 000 bytes; total user-accessible capacity may vary depending on operating environment. When referring to DDM capacity, gigabyte (GB) means 1 000 000 000 bytes; total user-accessible capacity may vary depending on the operating environment.

The ESS Model E10 does not support an expansion enclosure. ESS Models F10 and F20 have a faster processor than the ESS Models E10 and E20. They also offer more cache, 8 GB or 16 GB.

---

<sup>1</sup> For processor storage, real and virtual storage, and channel volume, a terabyte equals 2<sup>(40)</sup> or 1 099 511 627 766 bytes. For disk storage capacity and communications volume, a terabyte equals 1 000 000 000 000 bytes.

<sup>2</sup> A gigabyte of storage is 10<sup>9</sup>bytes. A gigabyte of memory is 2<sup>(30)</sup> bytes.

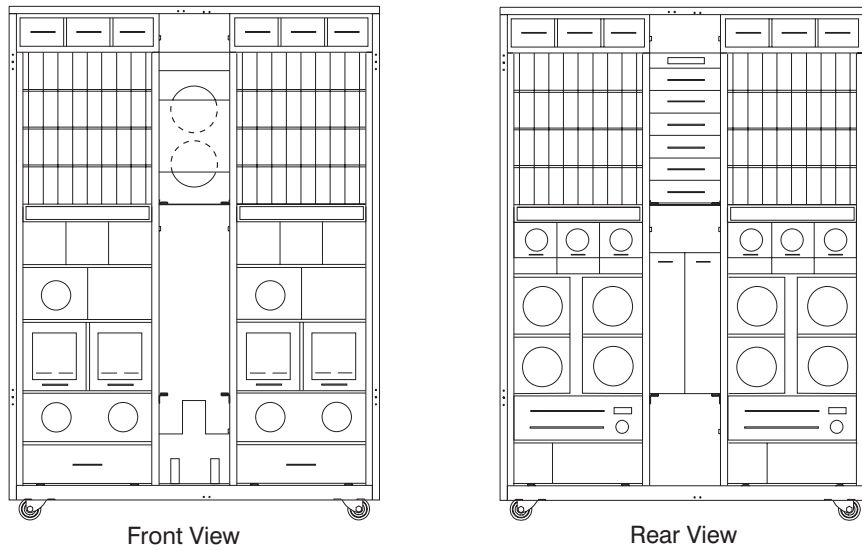


Figure 1. Models E10, E20, F10 and F20 front and rear views

Each ESS expansion enclosure has dual power cables and redundant power. The redundant power system allows both the storage controller and DDMs to continue normal operation when one of the power cables is inactive. Redundancy also ensures continuous data availability.

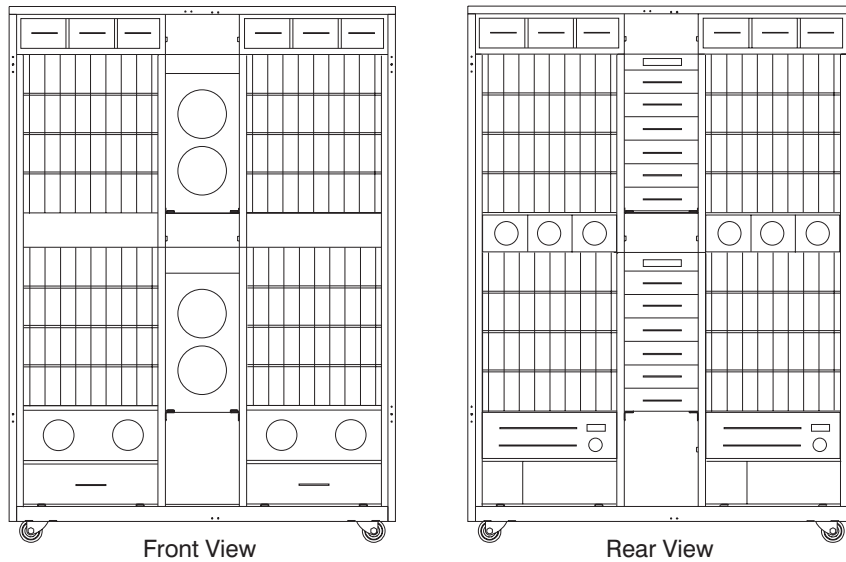


Figure 2. Expansion enclosure, front and rear views

The ESS Model E10 supports a maximum of 64 DDMs. The ESS Model E20 supports a maximum of 128 DDMs. The Model E20, with the expansion enclosure, supports a maximum of 384 DDMs.

---

## Host systems supported by the IBM ESS

The following sections contain information for attaching your ESS to:

- Open systems (SCSI attachment)
- Short wave fibre-channel attachment to an open host system
- S/390

## SCSI host systems

The ESS provides heterogeneous data storage that you can share with open host system (SCSI and fibre-channel attachment) and System/390 work loads. The ESS supports the following interfaces and host systems. With SCSI adapters you can connect up to 128 host systems (four per SCSI interface).

The following host systems support SCSI attachment:

**Note:** See the Web site <http://www.ibm.com/storage/ess/ess.htm> for details about the types, models, adapters and operating systems supported for SCSI host systems.

- Hewlett Packard (HP-UX operating system)
- IBM RISC System/6000™ and RISC System/6000 SP™ (IBM AIX™ operating system)
- IBM AS/400 (IBM Operating System/400 Version 3 or Version 4 (OS/400®) operating system)
- IBM Netfinity® and Intel-based PC servers (Microsoft™ Windows NT™ operating systems)
- Sun (Solaris operating system)
- Data General (DG UX operating system)
- Intel-based PC servers (Novell Netware)
- Compaq AlphaServers (TRU64 UNIX and OpenVMS)

## Fibre-channel host systems

Each fibre-channel adapter has one port and can support one host.

The following host systems support short wave fibre-channel attachment to an open host system:

**Note:** See the Web site <http://www.ibm.com/storage/ess/ess.htm> for details about the types, models, adapters and operating systems supported for fibre-channel host systems.

- Hewlett Packard (HP-UX operating system)
- IBM RISC System/6000 and RISC System/6000 SP (IBM AIX operating system)
- IBM Netfinity and Intel-based PC servers (Microsoft Windows NT operating systems)
- Sun (Solaris operating system)
- Data General (DG UX operating system)
- Intel-based PC servers (Novell Netware)

- Compaq AlphaServers (TRU64 UNIX and OpenVMS)

## S/390 host systems

With ESCON adapters, you can have up to 32 connections, each with up to 64 logical paths.

The following S/390 ESCON models support the following operating systems:

**Note:** See the Web site <http://www.ibm.com/storage/ess> for details about the functional modes and software levels supported for S/390.

- MVS™
- VM
- VSE
- TPF

---

## Web interfaces

This section describes the Web security, the ESSNet, and the Web interfaces for ESS. The Web interfaces include:

- StorWatch Enterprise Storage Server Specialist (ESS Specialist)
- StorWatch Enterprise Storage Server Specialist Copy Services (ESS Copy Services), an optional ESS feature
- StorWatch Enterprise Storage Server Specialist Expert (ESS Expert), an optional software product

See your IBM field office for information about how to order the optional Web interfaces. See the *IBM Enterprise Storage Server Web Interface User's Guide* for detailed descriptions of the ESS Specialist and the ESS Copy Services Web interface and instructions about how to use the interfaces.

## Web connection security

You connect to the ESS through the ESSNet.

All data that is sent between the ESS and the Web browser through the ESSNet is encrypted to avoid unauthorized modification of configuration commands. Passwords and authorization levels protect access to the interface.

You control user access by assigning levels of access and passwords.

## IBM Enterprise Storage Server Network

The IBM Enterprise Storage Server Network (ESSNet) is a private network resident on an IBM workstation. It is a required feature. IBM installs the ESSNet when they install your IBM Enterprise Storage Server. ESSNet hardware includes:

- The IBM workstation (a PC) and monitor
- An external Ethernet hub that provides cable connections from the ESSNet to the IBM Enterprise Storage Server

**Note:** You can attach your Ethernet LAN to the external hub. You provide any hardware that is needed for this connection.

- A modem and a modem expander that allow your ESS to communicate between you and IBM

**Note:** Remote service support includes this equipment.

The ESSNet software on the workstation includes:

- Windows NT 4.0 operating system
- Browser software (Microsoft Internet Explorer) that allows access to the ESS Specialist
- The ESSNet application for installation and configuration. The ESSNet workstation includes an application that provides links to the ESS. Clicking one of these links initiates the ESS Specialist

The ESSNet provides:

- Support for multiple ESSs. A hub with 16 ports will support 7 ESSs.
- Connectivity between your ESS and the ESS Specialist Web interface. The ESSNet provides browser software at the appropriate level for this connection.
- Improvements in Web performance compared to other Web browser connections.
- Faster network connectivity and elimination of network setup problems when using other Web connections.
- Ethernet connectivity through an Ethernet hub to the ESSNet.
- An independent platform that facilitates installation and configuration of the ESS.
- Software for maintenance and configuration.

IBM installs the ESSNet when they install your first IBM Enterprise Storage Server.

## Accessing ESS Specialist and ESS Copy Services

You access the IBM StorWatch Enterprise Storage Server Specialist (ESS Specialist) and IBM StorWatch Enterprise Storage Server Copy Services (ESS Copy Services) from the ESSNet. The ESSNet includes browser software for this access.

You access ESS Copy Services from ESS Specialist.

You can access ESS Specialist from your own workstation if you connect the ESSNet into your intranet using one of the ports on the ESSNet hub. Access ESS Specialist using a browser that is fully enabled for Java™ 1.1, such as Netscape Navigator 4.5 or Microsoft Internet Explorer 4.

If you do not choose to attach ESSNet into your intranet, then you can only access ESS Specialist from Microsoft Internet Explorer on the ESSNet workstation (an IBM personal computer).

You must provide another network connection to the Remote ESS for peer-to-peer Remote Copy Web Copy Service support.

## ESS Specialist

The ESS includes the ESS Specialist. The ESS Specialist is a Web-based interface that allows you to configure the ESS.

From the Web interface, you can perform the following tasks:

- Monitor error logs
- View and change the configuration:
  - Add or delete SCSI and fibre-channel host systems
  - Configure SCSI and fibre-channel host ports on the ESS
  - Define controller images for System/390
  - Define fixed-block (FB) and count-key-data (CKD) disk groups
  - Add FB and CKD volumes
  - Assign volumes to be accessible to more than one host system
  - Change volume assignments
- Change and view communication resource settings, such as e-mail addresses and telephone numbers
- Authorize user access

With the ESS Specialist, you can view the following information:

- The external connection between a host system and an ESS port
- The internal connection of SCSI ports to cluster 1 or cluster 2
- How the storage space is allocated to FB and CKD volumes

## ESS Copy Services

Copy Services provides a Web-based interface for managing peer-to-peer remote copy (PPRC) and FlashCopy commands. Copy Services collects information from your IBM storage servers on a single Copy Services server.

Copy Services is part of the IBM Enterprise Storage Server Copy Services Web interface. You access Copy Services from the ESS Specialist main menu.

Use the Copy Services panels to view and define:

- **Volumes**

The Volumes panel allows you to view volumes and define them as source or target volumes for the PPRC and FlashCopy.
- **Server**

The Server panel allows you to work with logical servers as complete entities. You can build tasks to place the volumes of a logical server within a peer-to-peer relationship with the volumes of another logical server.

You can also build a task to remove similar groups of volumes from an existing peer-to-peer relationship.
- **Paths**

The Paths panel displays the current status of paths between one physical server and the servers to which it is connected.



You can also use this panel to add or remove copy service paths.

- **Tasks**

The Tasks panel allows you to manage tasks you have defined. You can run, remove, export, or import tasks.

- **Configuration**

The Configuration panel allows you to add to or save the existing configuration. You can also use this panel to display the error log.

## ESS Expert

The IBM StorWatch Enterprise Storage Server Expert (ESS Expert) is an optional software product that you can purchase to use with the ESS. The ESS Expert Web interface provides storage-resource management for the IBM storage servers.

- **Asset management**

ESS Expert collects and displays asset management data.

- **Capacity management**

ESS Expert collects and displays capacity management data.

- **Performance management**

ESS Expert collects and displays performance management data. For example:

- Number of I/O requests
- Number of bytes transferred
- Read and write response time
- Cache usage statistics

ESS Expert allows you to schedule the information collection. With this information, you can make informed decisions about volume placement and capacity planning as well as isolate I/O performance bottlenecks.

---

## Service interface

The ESS provides service interface ports for external connection of a service terminal.

IBM or your service provider can perform service on the ESS by using an IBM mobile service terminal (MoST) or equivalent.

## Remote services support

The ESS service interface also provides remote service support with call-home capability for directed maintenance for service personnel.

You provide an analog telephone line to enable this support. The service interface provides an RS232 connection through a modem switch and modem to the analog telephone line.

You need to order a modem and modem switch. The first ESS you order requires this equipment. The modem and modem switch support up to seven ESSs. The cable length from the ESS to the modem switch should be a maximum of 15 m (50 ft).

The ESS provides the following service functions:

- Continuous self-monitoring initiates a call (call home) to service personnel if a failure has occurred. Because service personnel who respond to the call know about the failing component, the service personnel can reduce the repair time.
- Error and problem logs are available that service personnel can access remotely to analyze potential failures.
- Remote support allows the ESS to correct many types of problems. When the ESS reports a problem, service personnel can often create a correction and apply it from the remote location.

---

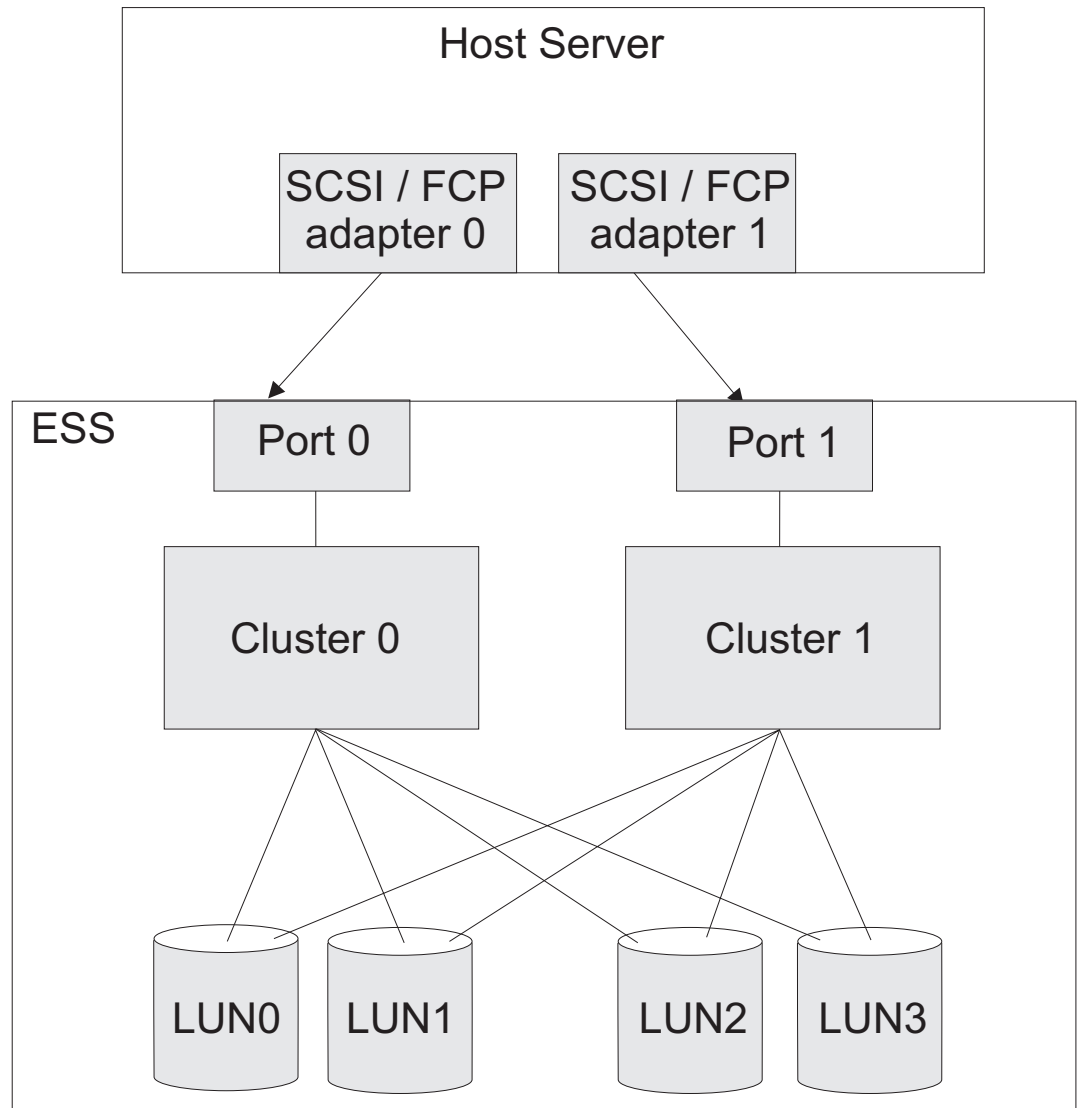
## Configuration

IBM logically configures your ESS during installation. After IBM installs the ESS, you can perform additional configuration by using the ESS Web interfaces. You can also change the remote service functions.

---

## Introducing the IBM Subsystem Device Driver

The IBM Subsystem Device Driver resides in the host server with the native disk device driver for IBM Enterprise Storage Server (ESS). It uses redundant connections between the host server and disk storage in an ESS to provide enhanced performance and data availability. These connections comprise many different components through which data flows during input and output processes. Redundancy and the ability to switch between these components provides many different paths for the data to travel. In the event of failure in one input-output path, it automatically switches to another input-output path. This automatic switching in the event of failure is called *failover*.



S009000Q

Figure 3. Multi-path connections between host server and ESS LUNs

The IBM Subsystem Device Driver provides the following functions:

- Enhanced data availability
- Automatic path failover and recovery to an alternate path
- Dynamic load balancing of multiple paths
- Setting path selection policies (AIX only)
- Concurrent download of licensed internal code

In most cases, host servers are configured with multiple host adapters with SCSI or fibre-channel connections to an ESS that, in turn, provides internal component redundancy. With dual clusters and multiple host interface adapters, the ESS provides more flexibility in the number of input-output (I/O) paths that are available. When there is a failure, the IBM Subsystem Device Driver reroutes I/O operations from the failed path to the remaining paths. This prevents a bus adapter on the host server, an external SCSI cable, a fiber connector, a cluster on the ESS, or a host interface adapter on the ESS from disrupting data access.

In addition, multipath load balancing of data flow prevents a single path from becoming overloaded with I/O.

The IBM Subsystem Device Driver path recovery algorithms have also been improved. The IBM Subsystem Device Driver will now test **Dead** paths that are not manually offline every 30 minutes. If the test is successful, paths will be brought back online sooner than they have been previously. The improved algorithms also detect failing paths earlier.

Concurrent download of licensed internal code is the capability to download and install licensed internal code on a ESS while applications continue to run. During the time when new licensed internal code is being installed in an ESS, the upper interface adapters inside the ESS may not respond to host I/O requests for approximately 30 seconds. The IBM Subsystem Device Driver makes such failures transparent to the host through its path selection and retry algorithms.

**Note:** For updated and additional information not included in this publication, see the README file on the IBM Subsystem Device Driver compact disk recordable (CDR) or visit the Subsystem Device Driver web site at:  
[www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates](http://www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates) °

---

## Chapter 2. Elements of data access

The ESS storage subsystem emulates the characteristics of the physical device. References to devices in this section and all sections in this publication which are concerned with the program's view of data are actually referring to the emulated volumes rather than the physical devices.

The basic elements of Data Access are described here as they appear to the user of the CCW interface to devices and storage directors. For example, the count area of a record, as seen by a program through this interface, consists of eight bytes; a five-byte record ID, a one-byte key area length, and a two-byte data area length. The count area actually recorded on the device, however, contains a flag byte, a physical address, and a cell address.

Data accesses involving only the cache and not the device will have the same characteristics as access from the device.

---

### Data groupings

The basic unit of data recorded on the device is the eight bit *byte*.

Bytes are combined to form an area which is the basic unit of data transfer between the device and the host system.

Areas are combined to form a record, which is the smallest *addressable* unit of data stored on the device.

One or more records may be written on a *track*, which is the smallest *directly* addressable unit of storage space on the device. A track is the smallest unit of storage space that can be independently formatted or protected from unauthorized access.

**Note:** The term record, as used in this document, is consistent with its accepted usage in CKD product description documents. It should be noted that this usage is not necessarily consistent with the traditional programming notion of a record.

The track is necessarily the minimum unit of space allocation when formatting write operations are permitted, or when the storage director is required to enforce access protection.

---

### Record format

A record consists of three areas: a count area, a data area, and an optional key area. Records are self-describing because the count area of each record contains information that specifies the format of the record.

## Count area

The count area is eight bytes long and consists of three fields which identify the record and define the lengths of the key and data areas. The count area is written at the time the record is originally written (formatted) and is not changed unless the record is re-formatted.

The count area has the following format:

Bytes	Meaning
0-4	Record ID
5	Key length (KL)
6-7	Data length (DL)

The *Record ID* is a five-byte field that is used to identify a record. There is no prescribed format or content for the record ID. Unique record IDs are required to assure access to the correct record.

**Note:** Identical record IDs may be used for special applications but these IDs should not be used as search arguments.

**Note:** By convention, MVS data management programming assigns unique record IDs of the form 'CCHHR' where 'CCHH' is the track address (see "Track address" on page 13) and 'R' is the sequential number of the record on the track.

VM also has a convention for assigning record IDs. It assigns unique record IDs of the form 'CCHHR' where 'CC' is the cylinder address relative to the start of the mini-disk and 'HH' is the head address (see "Track address" on page 13) and 'R' is the sequential number of the record on the track.

The *key length* is an unsigned eight-bit binary number which defines the length, in bytes, of the key area. The key length may have any value from 0 to 255. A value of zero specifies that the record has no key area.

The *data length* is an unsigned sixteen-bit binary number which defines the length, in bytes, of the data area. The data length may have any value from 0 to 65,535. A value of zero defines the record to be an end-of-file record (see "End-Of-File" on page 18). Individual device implementations limit the data area length to less than 65,535 bytes. See Appendix A, "Device data, geometries, and facilities" on page 219 for individual device type limitations.

## Key area

The key area contains information which may be used to identify the record.

The length of the key area is defined in the count area when the record is formatted and cannot be changed until the record is re-formatted. The contents of the key area may be altered without reformatting the record. The CCW operation codes, supported by this subsystem, that alter the contents of the key area of a record will always alter the contents of the data area of that record, if the data area exists. Therefore, the contents of the key area of a record cannot be altered without altering the contents of the data area of that record.

## Data area

The user data content of the record is recorded in the data area.

The length of the data area is defined in the count area when the record is formatted and cannot be changed until the record is re-formatted. The contents of the data area may be updated without affecting the count or key areas.

---

## Track format

The smallest directly addressable recording area on the device is a track. Each track has a unique address.

The first record on a track is a special single-area record called Home Address ("Home Address - HA" on page 14). Home Address is followed by another special record called Record Zero ("Record Zero - R0" on page 14). One or more user data records may be written on a track following Record Zero. The actual number of user data records that can be recorded on a track is a function of both the record length and the characteristics of the specific device on which they are recorded.

## Track address

Tracks are grouped into sets called cylinders. Each cylinder contains an equal number of tracks. Cylinders are numbered consecutively, starting with 0. Tracks within each cylinder are numbered consecutively, starting with 0.

A specific track is addressed by a two-level address of the form 'CCHH', where 'CC' is an unsigned 16-bit binary cylinder number, and 'HH' is an unsigned 16-bit binary track number within the cylinder.

## Functional track groups

The tracks are divided into functional groups. The range of track addresses assigned to each functional track group is defined in Appendix A, "Device data, geometries, and facilities" on page 219.

The functional track groups are described below with the access authorization required to access the tracks. First the access authorization for ECKD Architecture is described (which is applicable to 3390) followed by the deviations from the architecture for device types. Access authorization is described in the description of the Mask byte in "Mask Byte" on page 24.

The following table shows the functional track groups that may be accessed for each device type and access authorization.

*Figure 4. Access authorization for track groups*

---

Device	Access Authorization		
	Normal	Diagnostic	Device Support
3390	Pri	Diag	All

---

ESS does not implement alternate, diagnostic or Device Support tracks. Any attempt to access these tracks will be terminated with status that includes unit check. The default file mask is 'normal' access authorization.

## Primary tracks

Primary tracks are those tracks that contain normal data records for access by system and user programs.

Primary tracks may be addressed by channel programs that are executing with Normal or Device Support access authorization.

## Index

Index is a unique point or condition that defines the start of the track. Each addressable track has one and only one Index.

## End-of-Track

End-of-Track is a unique point or condition that defines the end of the track. Each addressable track has one and only one end-of-track point.

## Home Address - HA

Home Address (HA) is a special five-byte, single-area record that identifies the track and defines its operational status.

Special commands are provided to read and write Home Address. When Home Address is written, all records on the track, including Record Zero, are erased.

## Record Zero - R0

The first record on a track, following Home Address, is Record Zero (R0).

Special read and write commands are provided to read and format Record Zero. Whenever Record Zero is re-formatted, all user data records on the track are erased.

For ESS the key length of Record 0 is always zero and the data length is always eight.

---

## Data access

Access to data recorded on a track requires a physical access to the track (or the image of the track in cache), orientation to a recognizable point or recorded object on the track, and, for some functions, an associative addressing (search) operation that locates a specific record on the track. Each of these operations is described in the following sections.

## Track accessing

Each command chain must include a command that specifies a 'CCHH' before a Seek Head command or any command requiring track data access is accepted. Commands that specify a 'CCHH' are Seek, Seek Cylinder, Locate Record, Recalibrate, and Read IPL.

Track access may be accomplished by the execution of a Locate Record command, a Read IPL command, one of the seek commands, or by the track switching function of a multi-track operation (see "Multi-track operations" on page 22).



Rotational Position Sensing permits the storage director to position the device more precisely with respect to the desired record before it establishes orientation. This involves directly addressing a particular displacement on the track, relative to Index. The address element of this positioning operation is a *sector*.

Sector positioning on a rotating device is considered cyclic rather than static positioning because the device cannot remain positioned at a sector as it can to a track; it repositions itself to the sector once every rotation of the storage medium.

Sector positioning is accomplished by execution of the Locate Record command or the Set Sector command.

## Orientation

To access data recorded on the track, the storage director must first establish orientation, which is a state of awareness of the type of area that has most recently passed the read/write mechanism. If no such state of awareness exists, the storage director is said to be unoriented.

Orientation is established by locating either the starting point of the track (Index), or the start of a count area other than the Record Zero count area. As channel program execution continues, the storage director maintains orientation, switching from one orientation state to another as it proceeds through successive areas and records.

The seven possible states of orientation are:

1. Index state - between Index and Home Address
2. Home Address state - between Home Address and the Record Zero count area
3. Count Area state - between a count area and the following key or data area
4. Key Area state - between a key area and the following data area
5. Data Area state - between a data area and the following count area or, if no count area follows, end of track.
6. Pre-Count Area state - immediately before a count area other than the Record Zero count area
7. Reset Orientation state - the storage director is unoriented

At the start of a channel program the storage director is in the Reset Orientation state. Execution of a command that requires the storage director to read or write recorded data will cause the storage director to orient to either:

- Index state, by locating the Index point, or
- Pre-Count Area state, by locating a count area (other than R0)

As long as operations that require the storage director to read or write track data are being processed, the storage director maintains orientation, switching from one orientation state to the next as it reads, writes, or skips areas as specified by the channel commands being executed. When a command that does not require the storage director to process track data is executed (e.g., an addressing or control command) the orientation state is reset. When a channel program ends, the orientation state is reset.

**Note:** Throughout this document, the phrase“oriented to ---” is used interchangeably with the phrase“in the --- orientation state”; for example,“oriented to a data area” has the same meaning as“in the data area orientation state”.

## Record accessing

A specific record may be located using an associative addressing (search) operation which requires the storage director to read some recorded area from the device and compare (associate) it with a value obtained from the channel.

The search operation is performed by the Locate Record command or one of the search commands.

---

## Data transfer

Data transfer commands are commands that transfer data that is, or is to be, recorded on the track. All data transfer commands except Read IPL and Read Multiple CKD may be executed in a Locate Record domain.

The Locate Record command must be preceded by a Define Extent or Read IPL command in the same channel program. Only one Define Extent command is permitted in a channel program; however, a channel program may contain more than one Locate Record command.

The functions of Define Extent and Locate Record are described briefly in the following sections; for a complete description of the commands and their parameters refer to the command descriptions in “Define Extent” on page 23, and “Locate Record” on page 29.

## Define Extent

The Define Extent command defines the extent in which the channel program will operate and defines certain attributes of, and limitations on, the commands that follow Define Extent in the channel program. It also defines usage and control attributes for cache management of the remainder of the channel program.

An extent is a set of consecutively addressed tracks which can be accessed by a channel program. The limits of an extent are defined by specifying the addresses of the first and last tracks in the extent.

## Locate Record/Locate Record Extended

There are two forms of the Locate Record command, both of which provide the same basic function. The Locate Record Extended command provides all of the functions provided by the base Locate Record command (except Read Data - 06) as well as an additional function. The base Locate Record command performs three functions in the channel program. It:

- Specifies the operations to be performed.
- Specifies the address of the first track to be operated on and the orientation state to be established before data transfer is initiated.
- Defines the bounds and attributes of an operational *domain*. This domain is defined to begin with the transfer of the Locate Record parameters and to extend through the last operation specified by the Locate Record parameters.

Some data transfer commands are executed differently within the operational domain of a Locate Record.

The Locate Record function is provided by either the Locate Record command or the Locate Record Extended command. Both commands perform the three functions described above. The Locate Record command uses a 16 byte parameter list. The Locate Record Extended command uses a variable length parameter list (for ESS the maximum length is 22 bytes) which is composed of the Locate Record command parameter list, plus additional parameters.

**Note:** Performance can be significantly affected by the way that Locate Record CCWs are used in the channel program. For example, two different channel programs might be constructed for the purpose of reading two consecutive records on a track. One channel program consists of a single Locate Record CCW followed by two read CCWs, the other consists of two Locate Record CCWs, each followed by a single read CCW. While the two channel programs will provide the same function, the second will *always* take longer in execution because, after transferring the first record, the storage director must transfer and validate another Locate Record parameter list and position the device to begin the transfer of the second record.

It should also be noted that, although there is no provision in Locate Record for transferring *non-consecutive* records in the domain of a single Locate Record command, there is a provision to read *non-consecutive* tracks in the domain of some Locate Record Extended operations.

## Formatting

User data records are formatted on a track by executing formatting write commands (for example, Write Count Key and Data or Write Count Key and Data Next Track). The formatting write commands establish a count area, and format the key and data areas of a record. These commands, if not immediately followed by another formatting write command, always erase the remainder of the track after writing the record.

Formatting write commands must be used to change the format of a record, that is, the length of the key or data areas of a record.

## Erase

A record is said to be erased when it is rendered incapable of being read by the commands and facilities normally used for reading recorded data from the device. The manner in which the erase operation is performed is overwriting the record referenced with pad characters and then erasing the remainder of the track.

## File protection

A file mask is defined to provide a means for protecting the integrity of data on the device from multiple users. The file mask limits the write operations that are permitted in a channel program. It may also be used to confine channel program execution to operations on a single track or within a single cylinder.

At the start of a channel program, the file mask is zero. When a Define Extent or Set File Mask command is executed, the file mask is set equal to the value in the

Mask byte parameter. The format of the file mask is described in the description of the Mask byte parameter of Define Extent (see “Mask Byte” on page 24).

## End-Of-File

A record with a data length of zero is an end-of-file (EOF) record.

When a read command, other than Read Track Image, Read Track, or Read Multiple Count Key and Data, processes the data area of an end-of-file record, no data transfer takes place and the operation is terminated with Channel End, Device End, and Unit Exception status. Update write commands executing outside the domain of a Locate Record or within a Locate Record with CKD Conversion mode active will also be terminated with Channel End, Device End, and Unit Exception status when they attempt to update the data area of an end-of-file record.

The size of an end-of-file record can be calculated with the track capacity formula for the device, using a data length of one byte.

## Chapter 3. Command descriptions

This section contains descriptions of the channel commands that are supported by this storage subsystem. Topics include:

- “Command summary”
- “Exception conditions” on page 22
- “Command operations” on page 22
- Command descriptions, organized under the command categories shown in Figure 5
- Numeric command list in Figure 6 on page 21

### Command summary

Figure 5 is a quick reference list of the ESS commands, grouped by functional category. Figure 6 on page 21 lists the commands numerically in ascending hexadecimal order. In both tables, each command is listed with its command name, mnemonic, hex code, and a page reference to the command description.

Figure 5 (Page 1 of 2). Channel commands listed by function

Command Name	Mnemonic	Hex Code	Page
<b>Addressing and Control Commands</b>			
Define Extent	DX	63	23
Locate Record	LR	47	29
Locate Record Extended	LRE	4B	41
Seek	SK	07	51
Seek Cylinder	SKC	0B	51
Seek Head	SKHD	1B	52
Set File Mask	SFM	1F	53
Set Sector	SS	23	53
Read Sector	RS	22	54
<b>Read Commands</b>			
Read Special Home Address	RSHA	0A	61
Read Home Address	RHA	1A (9A)	62
Read Record Zero	RR0	16 (96)	62
Read Count Key and Data	RCKD	1E (9E)	63
Read Key and Data	RKD	0E (8E)	64
Read Data	RD	06 (86)	65
Read Count	RC	12 (92)	66
Read Multiple Count Key and Data	RMCKD	5E	67
Read Track	RT	(DE)	68
Read IPL	IPL	02	70
<b>Search Commands</b>			
Search HA Equal	SHA	39 (B9)	71
Search ID Equal	SIDE	31 (B1)	72
Search ID High	SIDH	51 (D1)	73
Search ID Equal/High	SIDEH	71 (F1)	73
Search Key Equal	SKE	29 (A9)	73
Search Key High	SKH	49 (C9)	74
Search Key Equal/High	SKEH	69 (E9)	74
<b>Write Commands</b>			
Write Special Home Address	WSHA	09	75
Write Home Address	WHA	19	77
Write Record Zero	WR0	15	76
Write Count Key and Data	WCKD	1D	76
Write CKD Next Track	WCKDNT	(9D)	79
Erase	ERAS	11	80

Figure 5 (Page 2 of 2). Channel commands listed by function

Command Name	Mnemonic	Hex Code	Page
Write Update Key and Data	WUKD	(8D)	81
Write Key and Data	WKD	0D	82
Write Update Data	WUD	(85)	83
Write Data	WD	05	84
<b>Sense Commands</b>			
Sense	SNS	04	86
Sense ID	SNSID	E4	87
<b>Miscellaneous Commands</b>			
No-Operation	NOP	03	88
Read and Reset Buffered Log	RRBFL	A4	89
Diagnostic Control(1)	DCTL	F3	90
Read Configuration Data	RCD	FA	95
Read Device Characteristics	RDC	64	54
<b>Path Control Commands</b>			
Device Reserve	RES	B4	99
Device Release	REL	94	100
Unconditional Reserve	UR	14	101
Reset Allegiance	RSTA	44	102
Set Path Group ID	SPID	AF	104
Sense Path Group ID	SNID	34	105
Suspend Multi-path Reconnection	SMR	5B	106
<b>Subsystem Commands</b>			
Set Cache Storage Mode	SSM	87	107
Perform Subsystem Function	PSF	27	111
Sense Subsystem Status	SNSS	54	134
Read Subsystem Data	RSSD	3E	139
Read Message ID	RMID	4E	153

Figure 6. Numeric list of channel commands

Command Name	Mnemonic	Hex Code	Page
Read IPL	IPL	02	70
No-Operation	NOP	03	88
Sense	SNS	04	86
Write Data	WD	05	84
Read Data	RD	06 (86)	65
Seek	SK	07	51
Write Special Home Address	WSHA	09	75
Read Special Home Address	RSHA	0A	61
Seek Cylinder	SKC	0B	51
Write Key and Data	WKD	0D	82
Read Key and Data	RKD	0E (8E)	64
Erase	ERAS	11	80
Read Count	RC	12 (92)	66
Unconditional Reserve	UR	14	101
Write Record Zero	WR0	15	76
Read Record Zero	RR0	16 (96)	62
Write Home Address	WHA	19	77
Read Home Address	RHA	1A (9A)	62
Seek Head	SKHD	1B	52
Write Count Key and Data	WCKD	1D	76
Read Count Key and Data	RCKD	1E (9E)	63
Set File Mask	SFM	1F	53
Read Sector	RS	22	54
Set Sector	SS	23	53
Perform Subsystem Function	PSF	27	111
Search Key Equal	SKE	29 (A9)	73
Search ID Equal	SIDE	31 (B1)	72
Sense Path Group ID	SNID	34	105
Search HA Equal	SHA	39 (B9)	71
Read Subsystem Data	RSSD	3E	139
Reset Allegiance	RSTA	44	102
Locate Record	LR	47	29
Search Key High	SKH	49 (C9)	74
Locate Record Extended	LRE	4B	41
Read Message ID	RMID	4E	153
Search ID High	SIDH	51 (D1)	73
Sense Subsystem Status	SNSS	54	134
Suspend Multi-path Reconnection	SMR	5B	106
Read Multiple Count Key and Data	RMCKD	5E	67
Define Extent	DX	63	23
Read Device Characteristics	RDC	64	54
Search Key Equal/High	SKEH	69 (E9)	74
Search ID Equal/High	SIDEH	71 (F1)	73
Write Update Data	WUD	(85)	83
Set Cache Storage Mode	SSM	87	107
Write Update Key and Data	WUKD	(8D)	81
Device Release	REL	94	100
Write CKD Next Track	WCKDNT	(9D)	79
Read and Reset Buffered Log	RRBFL	A4	89
Set Path Group ID	SPID	AF	104
Device Reserve	RES	B4	99
Read Track	RT	(DE)	68
Sense ID	SNSID	E4	87
Diagnostic Control(1)	DCTL	F3	90
Read Configuration Data	RCD	FA	95

---

## Exception conditions

The command descriptions in this section include specifications for checking and reporting I/O Program Exceptions that are described in Chapter 5, "Sense conditions" on page 169.

Exception conditions are identified in the command descriptions by exception type and format or message codes.

In the absence of an explicit statement to the contrary, the priority of exception conditions is defined by the sequence in which the exceptions are described in each command description.

The highest priority I/O program exception is Command Reject, format X'01', Invalid Command. The command codes accepted by ESS are summarized in Figure 5 on page 19.

---

## Command operations

### Multi-track operations

Data transfer operations may be executed in multi-track mode; that is, when the operation continues past the end of the 'current' track, the storage director attempts to advance to the 'next' track and continue the operation as defined in this section. Multi-track mode is selected when:

- bit 0 of the data transfer command equals '1', or
- a Locate Record operation is explicitly multi-track

#### Operations within the domain of a Locate Record

Multi-track mode is applicable when bit 0 of the data transfer CCW equals '1'. When multi-track mode is specified, multi-track data transfer commands must be used in the Locate Record domain. See "Read Data (06)" on page 38 for information about multi-track operations in a Read Data domain with CKD Conversion.

When a valid End-of-Track condition is detected prior to data transfer between the storage director and the device, and the CCW operation code specifies multi-track mode and the operation is not a Read Any or Write Any, the operation continues as follows:

- If the next track is within the defined extent, execution continues on that track; if the current track is the last track of a cylinder, execution continues on the first track of the next cylinder if the next cylinder is a primary cylinder. If the next cylinder is not a primary cylinder, execution is terminated with status that includes unit check (Command Reject and File Protected).
- If the next track is not within the defined extent, execution is terminated with status that includes unit check (File Protected).
- If the next track is marked as defective (Home Address Flag byte, bits 6-7 = '10'), execution is terminated with status that includes unit check (Command Reject - format X'0B', Invalid Alternate Track Pointer).



If the operation is a 'Read Any' or 'Write Any', execution continues, through index, on the same track.

### **Operations outside the domain of a Locate Record**

Outside the domain of a Locate Record, multi-track mode for read and search commands is selected when bit 0 of the CCW operation code is '1'. A multi-track read or search command that orients to Index, or that detects End-of-Track while attempting to orient to a count area, will attempt to advance to the next track and continue the operation as follows:

- If the file mask inhibits all seek commands and multi-track operations (bits 3-4 = '11'), execution is terminated with status that includes unit check (File Protected).
- If the next track is outside the defined extent, execution is terminated with status that includes unit check (File Protected).
- If the current track is the last track in the cylinder, execution is terminated with status that includes unit check (End of Cylinder).

---

## **Addressing and control commands**

See Figure 5 on page 19 for a complete list of channel commands.

### **Define Extent**

The base Define Extent parameters consist of 16 bytes, which define limits on subsequent operations, provide a blocksize value, and specify caching control 'hints' for the channel program.

ESS will defer validation of Define Extent parameters until after device end has been presented. As a result commands following the Define Extent will be terminated for errors in Define Extent parameters.

A Define Extent may not be preceded in the same channel program by a Define Extent, Read IPL or Set File Mask command. If these conditions are violated, the command execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director requests 16 bytes of parameters from the channel; if the channel provides fewer than 16 bytes, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

The format of the parameter list and criteria for validity checking are described in "Define Extent parameters" on page 24. Parameters and reserved fields are checked for validity in the order that they appear in the parameter list, except that the Architecture Mode (bits 0-1) in the Global Attributes byte is checked first. If an exception condition exists in more than one parameter in the list, only the first detected exception is reported.

Channel End and Device End status is presented after the parameter list is received.

Execution of Define Extent resets orientation in the storage director.

## Define Extent parameters

The Define Extent parameter list has the following format:

Byte	Parameter
0	Mask byte
1	Global Attributes
2-3	Blocksize in bytes
4-5	Cache Fast Write Data
6	Global Attributes Additional
7	Global Attributes Extended
8-11	Beginning of Extent Address
12-15	End of Extent Address

**Mask Byte:** The Mask byte replaces the contents of the file mask and may be used to inhibit or control certain operations in the channel program.

Figure 7 (Page 1 of 2). Define Extent, Byte 0, Mask Byte

Bits	Value	Definition
0-1	Write Control: These specify the types of write operations that are to be permitted in the channel program.	
	00	Permit all write operations except Write HA and Write R0
	01	Inhibit all write operations
	10	Permit update write operations only
	11	Permit all write operations. If all writes are permitted, byte 1, bits 3-5 are ignored and bypass cache is forced.
2	0	Must be zero. If this bit is not 0, execution of the next command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
3-4	Seek Control: These bits specify the seek commands and multi-track head switching operations that are to be permitted outside the domain of a Locate Record. They have no effect on channel commands that are executed within the domain of a Locate Record.	
	00	Permit all seek commands
	01	Permit Seek Cylinder and Seek Head commands only
	10	Permit Seek Head commands only
	11	Inhibit all seek commands and multi-track operations
5-6	Access Authorization	
	00	Normal Authorization
	01	Device Support Authorization Certain restricted operations require this authorization.
	10	Diagnostic Authorization Certain operations are valid only in channel programs that are operating with Diagnostic access authorization.
	11	Device Support Authorization with Inhibit Data Check Correction/Retry: For ESS this is the same as Device Support Authorization.

Figure 7 (Page 2 of 2). Define Extent, Byte 0, Mask Byte

Bits	Value	Definition
7	PCI Fetch Mode	<p>When PCI Fetch mode is active, the channel program is terminated with status that includes unit check any time that:</p> <ol style="list-style-type: none"> <li>1. Channel command retry has been used to recover from an exception condition during a read operation, and</li> <li>2. At least one byte of data has been transferred to the channel.</li> </ol> <p><b>Note:</b> Some host programs (for example, PCI Fetch) access data being read before ending status for the read command has been presented. This early access to the data being read conflicts with the concept of channel command retry. Channel command retry may be used by storage subsystems to re-execute a read command when the subsystem has reason to doubt the validity of the data transferred to the channel. If the retry results in transferring valid data to the host system, the host program still needs to be notified that it may have accessed the data whose validity was in doubt. Therefore, if the PCI fetch mode bit in the file mask is '1', the read command that was retried is terminated with status that includes unit check. If retry was unsuccessful, the channel program will be terminated anyway.</p>
	0	Not PCI Fetch Mode
	1	PCI Fetch Mode

**Global Attributes:** This parameter defines global attributes of operations within the channel program. It has the following format:

Figure 8 (Page 1 of 2). Define Extent, Byte 1, Global Attributes

Bits	Value	Definition
0-1	Architecture Mode	<p>Bits 0-1 must be '11' to specify ECKD Architecture mode. If bits 0-1 are not '11', execution of the next command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).</p> <p>This storage subsystem supports the ECKD architecture only.</p>
	00-10	Reserved
	11	ECKD
2	CKD Conversion Mode	<p>The handling of certain conditions is modified during execution of a CKD Conversion mode channel program, as described in the description of the Write Data (01) Locate Record operation code on page 35 and the definition of the Read Data (06) Locate Record operation code on page 38.</p> <p><b>Note:</b> CKD Conversion mode is intended to be used by a host program routine that dynamically converts CKD channel programs to ECKD format. Use of this mode permits the storage director to relax certain ECKD Architecture requirements (for example, the requirement that the actual length of a record to be updated must equal the length specified in the Locate Record or Define Extent parameters or the requirement that only end-of-extent violations are recognized in a domain). This, in turn, allows the conversion routine to build a Locate Record prefix for an update write CCW when the actual record length is not known or to recognize an end-of-cylinder exception in a Read Data domain.</p>
	0	Not CKD conversion
	1	CKD conversion mode

Figure 8 (Page 2 of 2). Define Extent, Byte 1, Global Attributes

Bits	Value	Definition
3-5		Subsystem Operating Mode These bits specify a Subsystem operating mode. This attribute determines the cache management algorithms for the command chain.
	000	Normal Cache Replacement The cache is managed using ESS standard algorithms.
	001	Bypass Cache Tracks accessed by this I/O chain will be placed in the middle of the LRU list.
	010	Inhibit Cache Loading Tracks promoted into cache by this I/O chain will be placed in the middle of the LRU list.
	011	Sequential Access Tracks following the track accessed by the CCW chain will be promoted to the cache in anticipation of a near term requirement and tracks preceding the accessed track may be demoted.
	100	Sequential Prestage Tracks following the track accessed by the CCW chain will be promoted to the cache in anticipation of a near term requirement and tracks will be demoted by LRU.
	101	Record Access When within the domain of a Locate Record without CKD conversion mode set, only the records accessed by the channel program are promoted into cache.  When this is set Regular Data Format Mode (in Global Attributes Extended) must also be set, or this indication is ignored (Normal Cache Replacement is assumed).
	110 through 111	Reserved: These combinations are ignored and result in Normal Cache Replacement. They should not be used.
6		Cache Fast Write Data This bit on indicates that write operations can complete without updating the DASD track. Updating the nonvolatile media is not required.  Cache Fast Write Data is only valid with normal access authorization. If this bit is set and bits 5-6 of byte 0 is not '00', execution of the next command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).  <b>Note:</b> If Cache Fast Write Access is used, this flag must be set for both read and write accesses to the data to insure integrity of the data.
	0	Not Cache Fast Write Data access
	1	Use Cache Fast Write Data access
7		Inhibit DASD Fast Write This bit is ignored by ESS.

**Blocksize:** Bytes 2-3 specify the blocksize in bytes. This value is an unsigned 16-bit binary value that specifies the maximum number of bytes to be transferred for a single record in the channel program. A zero value is valid and is equivalent to specifying the maximum record length supported by the device. The maximum record length supported by the device is the maximum record length + 8. See Figure 63 on page 221 for the maximum record lengths of the different devices.

If the blocksize exceeds the 'usable track capacity', execution of the next command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

When a subsequent Locate Record command specifies an update write type operation and does not specify a Transfer Length Factor, the blocksize value must

be the *exact* data length (or sum of key and data lengths if the key area is to be updated) of each record in the Locate Record domain. See the description of the Write Data in “Locate record operations” on page 34.

The proper blocksize value may be either:

- The data length (DL),
- The key length (KL) plus DL, or
- The sum of DL and KL plus 8 (for the count area), depending on the areas of the record that are to be transferred.

**Identification Bytes:** Bytes 4-5 contain the Cache Fast Write Data Identification (CFWID). If Cache Fast Write Data access (byte 1, bit 6 =1) is specified and this field is zero, execution of the next command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If Cache Fast Write Data access is specified (byte 1, bit 6 = 1), this 2-byte ID must compare with the current subsystem CFWID. If it does not, execution of the next command is terminated with status that includes unit check (Command Reject, format F, message 6, Cache Fast Write Access Not Authorized).

**Global Attributes Additional:** This parameter provides for an extension to the Define Extent Global Attributes.

*Figure 9. Define Extent, Byte 6, Global Attributes Additional*

Bits	Definition
0-3	Reserved. Set to Zero.
4-7	Authorized MPLF Partition  Any MPLF Set Lock State or Unlock orders that follow this command must specify the same Partition ID. A value of zero prohibits any MPLF orders from following this command.  <b>Note:</b> The value in this field is ignored when the 3990 Model 6 is running in TPF Mode (TPF Mode is set in VPD).

**Global Attributes Extended:** This parameter defines Global Attribute Operations, in addition to those defined in the Global Attributes byte (byte 1). It has the bit settings shown in Figure 10.

Figure 10. Define Extent, Byte 7, Global Attributes Extended

Bits	Value	Definition
0-1	Subsystem Operating Mode	
	00	No special mode recognized
01	Regular Data Format Mode	
	<p>When the Regular Data Format Mode indicator is set, the subsystem assumes that the following conditions are met:</p> <ul style="list-style-type: none"> <li>• There are no key fields (all records have a key length of zero).</li> <li>• The records are a fixed size in the extent.</li> <li>• The data length of every record in the extent is the same as the data length that is specified in the BlockSize/Transfer Length Factor in the channel program.</li> <li>• The record IDs are of the form, CCHHR, where, <ul style="list-style-type: none"> <li>– CC: can be any number for the first cylinder in the extent, but subsequent cylinder numbers increase monotonically, in consecutive order.</li> <li>– CC is the same for all tracks in the cylinder.</li> <li>– HH: physical head number.</li> <li>– R: record numbers begin at 1 on each track and increase monotonically, in consecutive order.</li> </ul> </li> <li>• R0 is standard format (8 byte data field, no key field) and has a data field of all binary zeros.</li> <li>• All tracks, except the last, in the domain of a Locate Record are fully formatted with records that conform to the formatting rules stated in this list.</li> </ul>	
10	Special Search Key Mode	
<p>Byte 7, bits 0, 1, when set to '10', indicate that Search Key operations may be carried out in Special Search Key mode. Restrictions on the use of this mode are given below.</p> <p>By setting special search key mode, the software is responsible for ensuring:</p> <ol style="list-style-type: none"> <li>1. That any search CCW that is encountered be in a search TIC loop</li> <li>2. That the search TIC loop consists of: <ul style="list-style-type: none"> <li>• Optionally a single Read Count command (may be multi-track).</li> <li>• A single Search Key command (may be multi-track, Search Key Equal, Search Key High, or Search Key Equal or High).</li> <li>• TIC</li> <li>• Nothing else</li> </ul> </li> <li>3. Data length of search key CCW must equal key length for all records to be searched.</li> </ol> <p><b>Warning:</b> Failure to follow these rules when in Special Search Key mode may yield unpredictable and inconsistent results.</p>		
11	TPF Record Cache Mode	
<p>When these bits are set to '11' and the ESS operates in TPF Mode (TPF Mode is set in VPD), record cache with Multi-Path Lock Facility (RPQ 8B0178) operates as described in the <i>TPF Support RPQs, GA32-0134</i>.</p> <p>When the ESS operates in either Basic Operations Mode or Enhanced Operations Mode 1 and these bits are set to '11', the subsystem operating mode defaults to Regular Data Format Mode, as if these bits were set to '01'. Multi-Path Lock Facility operates as described in this manual. In addition to channel command descriptions, see Chapter 4, "Multi-Path Lock Facility" on page 155.</p>		
2-4	Reserved	
5	Regular Record Zero	
	0	No permission is granted to modify R0 unless the channel program has changed it.
	1	<p>Permission is granted to the subsystem to format write R0 with:</p> <ul style="list-style-type: none"> <li>• an ID = CCHHR, where CC = physical cylinder number, HH = physical head number, and R = 0</li> <li>• a key length of zero</li> <li>• a data length of eight</li> <li>• a data field containing all zeros.</li> </ul>
6-7	Reserved	

**Beginning of Extent Address:** Bytes 8-11 contain the address, 'CCHH', of the first track of the extent.

'CCHH' must be a valid track address for the device:

1. If the Mask byte specifies Normal access authorization (bits 5-6 = '00'), the 'CCHH' will only be accepted if it is a Primary track address.
2. If the Mask byte specifies Device Support access authorization (bits 5-6 = '01' or '11'), 'CCHH' may be any valid track address.

If the address is not valid, execution of the following command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). See Appendix A, "Device data, geometries, and facilities" on page 219 for valid track address ranges.

**End of Extent Address:** Bytes 12-15 contain the address, 'CCHH', of the last track of the extent.

'CCHH' must be equal to or greater than the 'CCHH' value specified by the Beginning of Extent Address in bytes 8-11 and must be a valid track address for the access authorization and device type as defined in the previous Beginning of Extent Address section.

If 'CCHH' is not a valid track address, execution of the following command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). See Appendix A, "Device data, geometries, and facilities" on page 219 for valid track address ranges. If 'CCHH' is less than the value in bytes 8-11, execution of the following command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

## Locate Record

The Locate Record function is provided by both the Locate Record command and the Locate Record Extended command. See "Locate Record/Locate Record Extended" on page 16.

Locate Record transfers 16 bytes of parameters from the channel. The parameters specify the location and number of records to be processed and the operation to be performed. Execution of Locate Record creates an operational domain as described in "Locate Record/Locate Record Extended" on page 16.

If a Locate Record command is received within the domain of another Locate Record or Locate Record Extended command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If Locate Record was not preceded by a Define Extent or Read IPL command in the same channel program, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director requests 16 bytes of parameters from the channel; if the channel provides fewer than 16 bytes, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

The format of the parameter list and criteria for validity checking are described in "Locate Record Parameters" on page 30. Parameters and reserved fields are checked for validity in the order that they appear in the parameter list, except that the Operation code (bits 2-7) in the operation byte is checked first. If an exception

condition exists in more than one parameter in the list, only the first detected exception is reported.

After a valid parameter list is received and verified:

- If the operation can be executed as a cache hit then Channel End and Device End are presented.

**Note:** On reads, Locate Record processing assumes that if the track is in cache that the operation is a hit. If some data required by the chain is not in cache then the command accessing that data will be Channel Command Retried and the data will then be retrieved from the device.

- If data from the device is required to complete the operation then
  1. Channel End is presented.
  2. Track and sector positioning of the device is initiated.
  3. Device End is presented when the positioning is completed.

Orientation is reset in the storage director when Locate Record is received. Execution of Locate Record establishes the orientation state specified by the operation byte parameter before any data is transferred to or from the device.

### Locate Record Parameters

The Locate Record parameter list has the following format:

Byte	Name
0	Operation byte
1	Auxiliary byte
2	Must be zero
3	Count
4-7	Seek address
8-12	Search argument
13	Sector number
14-15	Transfer length factor

If byte 2 is not zero, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

**Operation Byte:** Byte 0 specifies the operations to be performed within the Locate Record domain and the orientation to be established at the completion of track access.



Figure 11. Locate Record, Byte 0, Operation Byte

Bits	Value	Definition
0-1		Orientation These bits are orientation modifiers that specify the orientation to be established by the execution of Locate Record when the operation specified in bits 2-7 requires orientation to a specific record or record area before data transfer is initiated.
	00	Count area This orientation specifies that data access for the CCW that immediately follows Locate Record is to begin with the storage director oriented to the count area of the record that satisfied the search operation. That CCW may, therefore, process the key and data areas of the record that satisfied the search operation. Count area orientation is valid with all Operation codes.
	01	Home Address This orientation specifies that data access for the CCW that immediately follows Locate Record is to begin with the storage director oriented to Home Address. Home Address orientation is valid with the Orient (00), Format Write (03), Read Data (06), Read Tracks (0C), and Read (16) Operation codes only. If bits 2-7 contain any other operation code when bits 0-1 = '01', Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
	10	Data area This orientation specifies that data access for the CCW that immediately follows Locate Record is to begin with the storage director oriented to the data area of the record that satisfied the search operation. That CCW, therefore, will not process any areas of the record that satisfied the search operation. Data area orientation is valid with the Orient (00), Write Data (01), Read Data (06) and Read (16) Operation codes only. If bits 2-7 contain any other operation code when bits 0-1 = '10', Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
	11	Index This orientation specifies that data access for the CCW that immediately follows Locate Record is to begin with the storage director oriented to Index. Index orientation is valid with the Format Write (03) and Read (16) Operation codes only. If bits 2-7 contain any other operation code when bits 0-1 = '11', Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
2-7		Operation Code These bits specify the operation to be performed within the Locate Record domain. The Operation codes supported by this storage subsystem are listed below. If any other Operation code is specified, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). The operations are described in "Locate record operations" on page 34.
	X'00'	Orient
	X'01'	Write Data
	X'03'	Format Write
	X'06'	Read Data
	X'0B'	Write Track
	X'0C'	Read Tracks
	X'11'	Erase
	X'16'	Read

Figure 12 on page 32 illustrates the valid combinations of orientation and operation code values.

Figure 12. Locate Record Operation Byte Values

Operation Code	Orientation			
	Cnt	HA	Data	Index
Orient	00	40	80	n
Write Data	01	n	81	n
Format Write	03	43	n	C3
Read Data	06	46	86	n
Write Track	0B	n	n	n
Read Tracks	0C	4C	n	n
Erase	11	n	n	n
Read	16	56	96	D6

**Note:** n - Combination is not valid.

**Note:** All operations that can be performed in the domain of a Locate Record command, except Read Data (X'06'), can be performed in the domain of a Locate Record Extended command.

**Auxiliary Byte:** Byte 1 is the Auxiliary byte. It is used to validate optional fields in the Locate Record parameter list.

Figure 13. Locate Record, Byte 1, Auxiliary Byte

Bits	Value	Definition
0	0	Bytes 14-15 are unused
	1	Bytes 14-15 contain a Transfer Length Factor that overrides the block size specified by the Define Extent parameter.
1-6	0	Must be zero
	1	If any of these bits are '1', Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
7	0	No Read Count CCW is suffixed to the Locate Record domain
	1	A Read Count CCW is suffixed to the Locate Record domain.  When this bit is set, it specifies that the last data transfer CCW in the Locate Record domain will be a Read Count CCW. The Read Count CCW is a suffix to the Locate Record domain and the record it operates on is included in the count of records to be processed as specified by the Count parameter.  Read Count CCW suffixing is only valid for a Locate Record that specifies a Write Data (01), or Read (16) Operation code. If suffixing is specified (bit 7 = '1') for a Locate Record that specified any other Operation code, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

**Count:** Byte 3 is the Count parameter. The 'Count' determines the domain or region of influence of the Locate Record. In general, the Count parameter specifies the number of records or tracks to be operated on by data transfer commands that follow Locate Record. Specific interpretation of the Count parameter depends on the Operation code in byte 0 and is described in "Locate record operations" on page 34 for each Operation code.

If the Operation code is Orient (00), the Count must be zero; otherwise, the Count must be nonzero. If Read Count Suffixing is specified in a Locate Record, the count must be greater than 1. If the Count does not satisfy these requirements, the Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

**Seek Address:** Bytes 4-7 specify a seek address 'CCHH'.

The value in bytes 4-7 must be a valid track address for the device. If it is not, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). See Appendix A, "Device data, geometries, and facilities" on page 219 for valid track address ranges.

If the Seek Address is not within the extent defined by the preceding Define Extent command, the Locate Record is terminated with status that includes unit check (File Protected).

**Search Argument:** Bytes 8-12 specify a value to be used as a search argument for the Locate Record search operation that is described on page 34.

When the operation specified in byte 0 does not require orientation to a specific record, no search operation is performed and bytes 8-12 are ignored. When Home Address orientation is specified, byte 12 is ignored.

**Sector Number:** Byte 13 contains a Sector Number to which the device is to be positioned before the storage director establishes orientation.

The sector number must be within the range of valid sector numbers for the device. If the sector number is invalid, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). See Appendix A, "Device data, geometries, and facilities" on page 219 for the range of valid sector numbers.

A value of X'FF' is valid and specifies that sector positioning is not to be performed before establishing orientation.

**Transfer Length Factor:** When byte 1, bit 0 is '0', bytes 14-15 must contain zeros; if bytes 14-15 are not zero, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

When byte 1 bit 0 is '1', bytes 14-15 contain a Transfer Length Factor (TLF). The Transfer Length Factor must be non-zero; if it is zero, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If the Transfer Length Factor value is greater than the value specified (or implied) in the Define Extent Blocksize parameter, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

The storage director uses the TLF to determine the number of data bytes to be requested from the channel for each write CCW that follows a Locate Record that specified the Write Data (01) Operation code. The product of the value in bytes 14-15 and the Count parameter is used to determine the total number of bytes to be transferred by data transfer commands that are executed within the domain of a Locate Record that specified the Format Write (03), Write Track (0B), Read Tracks (0C), or Read (16) Operation codes.

The TLF value is not retained by the storage director after the expiration of the Locate Record domain.

If Locate Record does not specify a Transfer Length Factor, the storage director will use the value from the Define Extent Blocksize parameter for any required data transfer length calculation.

### **Locate record operations**

The following subsections describe the processing that occurs as a part of the execution of the Locate Record command, the Locate Record Extended command, and some processing that is related to the data transfer command execution that follows Locate Record.

Processing that occurs as part of the execution of the individual data transfer commands is described as part of the descriptions of those commands.

The description of each operation specifies the number and type of data transfer commands that are expected or permitted in the Locate Record domain. If any other command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If ESS presents normal ending status without unit check or Unit Exception and the channel drops chaining without Stacking Status or issuing a Halt I/O or a System or Selective Reset before the expected number of data transfer commands in the Locate Record domain have been received, an Incomplete Domain exception occurs.

**Note:** If truncation of the data transfer occurs (by either ESS or the Channel) and the SLI bit is not set in the CCW flags, the channel will drop chaining and an Incomplete Domain exception may be presented.

**Search Operation:** Certain Locate Record operations require orientation to a specific record or record area before data transfer is initiated. This specific orientation is accomplished by performing a search operation as described below:

- When index orientation is specified the device is positioned to index, but no search operation is performed.
- When count or data area orientation is specified (byte 0, bits 0-1 = '00' or '10'), the value in bytes 8-12 of the Search Argument is compared with the record ID field in the count area of each record, including Record Zero, encountered on the track, after track orientation is established. Once started, the search operation continues until an equal compare occurs, or until all record IDs on the track have been processed. If no record ID on the track compares equal to the Search Argument, the operation is terminated with status that includes unit check (No Record Found).

If the track specified in the Seek Address parameter is marked as defective (HA Flag byte, bits 6-7 = '10'), execution is terminated with status that includes unit check (Command Reject - format X'0B', Invalid Alternate Track Pointer).

- When Home Address orientation is specified (byte 0, bits 0-1 = '01'), the value in bytes 8-11 of the Search Argument is compared with the track address field in the Home Address on the accessed track; byte 12 is ignored. If no Home Address is detected on the track, execution is terminated with status that includes unit check (the sense data is the Data Check appropriate for an 'HA, No Sync Byte' for the device type). If the comparison is unequal, the operation is terminated with status that includes unit check (No Record Found).

The Home Address search operation is performed on the track specified in the Seek Address parameter, even if that track is marked as defective.

**Orient (00):** This Operation code prepares the storage director to position the device in accordance with the Seek Address and Sector Number parameters.

Track access is initiated using the Seek Address and Sector Number parameters.

When Locate Record is followed by a search or read command, the search operation specified by the Search Argument and the Orientation modifiers (byte 0, bits 0-1) is performed after track access is completed. When Locate Record is not followed by a read or search command, the search operation is not performed. The search operation does not satisfy the search prerequisite for a write command.

The Locate Record domain expires with the completion of the track access or search operation.

**Write Data (01):** This Operation code prepares the storage director to update write one or more consecutive user data records, starting with the record specified by the Search Argument parameter. The number of records to be updated is indicated by the Count parameter. Write Data can also be used to update the data area of Record Zero when the search argument is set equal to the record ID in the R0 count area.

If the file mask inhibits all write operations (bits 0-1 = '01'), Locate Record is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the Define Extent command specified CKD Conversion mode (Global Attribute byte, bit 2 = '1'), the Locate Record must specify Count orientation; otherwise, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If the Define Extent command specified CKD Conversion mode (Global Attribute byte, bit 2 = '1'), only one write command is allowed in the domain. If Read Count Suffix is specified with CKD Conversion, the count must be 2; otherwise it must be 1. If this condition is not met, the Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

Track access is initiated using the Seek Address and Sector Number parameters.

When track access is completed, the search operation specified by the Search Argument and the Orientation modifiers (byte 0, bits 0-1) is performed.

When Auxiliary byte, bit 7 = '0', Locate Record must be followed by the number of update write commands specified by the count parameter; when Auxiliary byte, bit 7 = '1', Locate Record must be followed by a number of update write commands that is one less than the value specified by the Count parameter, and the last update write command must be followed by a multi-track Read Count command.

When count orientation is specified and the required number of update write commands is 1, Locate Record must be followed by a Write Data, Write Update Data, Write Key and Data, or Write Update Key and Data command; when the number of update write commands is greater than 1, Locate Record must be followed by a sequence of either Write Update Data or Write Update Key and Data

commands. Write Update Data and Write Update Key and Data commands may not both occur in the domain of the same Locate Record.

When data orientation is specified, Locate Record must be followed by a sequence of either Write Update Data or Write Update Key and Data commands. Write Update Data and Write Update Key and Data commands may not both occur in the domain of the same Locate Record.

If an invalid command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If no user data record is detected on the track, execution is terminated with status that includes unit check (No Record Found).

If a multi-track operation continues after the last record on a track has been updated, the storage director attempts to advance to the next track as described in "Operations within the domain of a Locate Record" on page 22 and update the first user data record on the next track. If the next track does not contain a user data record, execution is terminated with status that includes unit check (No Record Found).

The Transfer Length Factor (or the Blocksize from Define Extent, if no Transfer Length Factor was specified) must be equal to the actual record length of the records to be updated. The actual record length is either the data length or the sum of the key and data lengths, depending on whether the following write commands specify Write Data or Write Key and Data.

If the actual length of any record to be updated differs from the value in the TLF (or Blocksize) and CKD Conversion mode is not specified (Define Extent Global Attributes byte, bit 2 = '0'), execution is terminated with status that includes unit check (Invalid Track Format) and the record is not updated.

If the actual length of any record to be updated differs from the value in the TLF (or Blocksize) and CKD Conversion mode is specified (Define Extent Global Attributes byte, bit 2 = '1'), then:

- If the actual length of the record is zero, no data area transfer to the device is attempted and execution is terminated with status that includes Unit Exception.
- If the actual length of the record is not zero, then the TLF check is ignored and data transfer is completed.

**Format Write (03):** This Operation code prepares the storage director to format one or more consecutive user data records. The number of records to be formatted is specified by the Count parameter. Format Write may also be used to format Home Address and/or Record Zero when Index or Home Address orientation is specified.

If Home Address orientation is specified (byte 0, bits 0-1 = '01'), the file mask must permit writing HA and R0. If Index orientation is specified (byte 0, bits 0-1 = '11'), the file mask must permit writing HA and R0 and must specify Device Support Access authorization. If the file mask does not permit writing HA and R0 (bits 0-1 not = '11'), Locate Record is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). If the file mask does not specify Device Support Access authorization (bits 5-6 not = '01' or '11'),

Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If count area orientation is specified (byte 0, bits 0-1 = '00'), the file mask must permit format write operations. If the file mask does not permit format write operations (bits 0-1 = '01' or '10'), Locate Record is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If Index or Home Address orientation is specified and the track specified in the seek address has pinned data, then that track will be discarded, and the operation will proceed.

Track access is initiated using the Seek Address and Sector Number parameters. If Index or Home Address orientation is specified, a Sector value of zero is used for track access; the Sector Number parameter is not used.

When track access is completed, the storage director orients the device according to the Orientation modifiers (byte 0, bits 0-1). See "Search Operation" on page 34.

Locate Record must be followed by the number of formatting write CCWs specified by the Count parameter. Command sequence requirements for formatting write commands following Locate Record are as follows:

- When Index orientation is specified, Locate Record may only be followed by a Write Special HA CCW.
- When Home Address orientation is specified, Locate Record may only be followed by a Write HA or Write R0 CCW.
- When count area orientation is specified, Locate Record may only be followed by a Write CKD CCW.
- A Write Special HA or Write HA CCW may only be followed by a Write R0 CCW.
- A Write R0 CCW may only be followed by a Write CKD CCW.
- A Write CKD CCW may only be followed by a Write CKD or Write CKD Next Track CCW.
- A Write CKD Next Track CCW may only be followed by a Write CKD or Write CKD Next Track CCW.

If any other command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the storage director determines that there is not enough space on the track to write a record, the operation is terminated with status that includes unit check (Invalid Track Format).

**Programming Note:** The residual count in the SCSW<sup>3</sup> will not necessarily indicate the amount by which the record length exceeded the available space on the track.

After the last record is written on each track, the rest of the track is erased.

---

<sup>3</sup> Sub-Channel Status Word, see *Enterprise Systems Architecture/390, Principles of Operation*.

A Write Count, Key, and Data command that follows a Write Record Zero command to a track that has been marked defective, will be terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

**Read Data (06):** This Operation code prepares the storage director to transfer one or more records (which may include Home Address and Record Zero) to the channel. The number of records to be read is specified by the Count parameter.

Track access is initiated using the Seek Address and Sector Number parameters.

When track access is completed, the search operation specified by the Search Argument and the Orientation modifiers (byte 0, bits 0-1) is performed.

Locate Record must be followed by the number of read CCWs required to process the number of records specified by the Count parameter. The Read Multiple Count Key and Data, Read Track, Read Special HA, and Read IPL CCWs are not valid in the Locate Record domain; any other sequence of read commands is valid. If an invalid command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the operation continues past the end of a track, execution continues according to the state of bit 0 in the read CCW. If bit 0 is '0', the operation continues through index on the same track as defined for the applicable read command. If bit 0 is '1' multi-track operation is applicable and the operation depends on whether or not CKD Conversion mode is specified (Define Extent parameter byte 1 bit 2 = '1'). If CKD conversion is specified, multi-track operations are not allowed to continue beyond the end of the last track in a cylinder. In the CKD conversion mode case, if the next track is within the defined extent, execution is terminated with status that includes unit check (End of Cylinder); otherwise, execution is terminated with status that includes unit check (File Protected). If CKD conversion is not specified, the operation continues as described in "Operations within the domain of a Locate Record" on page 22.

**Write Track (0B):** This Operation code prepares the storage director to update write the data area of Record Zero and format the remainder of the track. The number of records to be formatted is one less than the value specified by the Count parameter. If the value in the Count parameter = 1, the remainder of the track is erased after the Record Zero data area has been updated.

If the file mask does not permit format write operations (bits 0-1 are '01' or '10'), Locate Record is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Track access is initiated using the Seek Address parameter and a Sector value of zero; the Sector Number is not used.

When track access is completed, the storage director compares the Search Argument with the record ID field in the Record Zero count area. If no Record Zero is found on the track, or if the Search Argument compares unequal to the record ID field in the Record Zero count area, the operation is terminated with status that includes unit check (No Record Found).



Locate Record must be followed by one Write Data CCW, and a number of Write CKD CCWs that is one less than the value in the Count parameter with the following exception:

1. An Erase may be substituted for the last or only Write CKD command in the domain.
2. If the Count parameter contains a value of 1, no Write CKD CCWs can follow the update data CCW. In this case the remainder of the track will be erased after the R0 data area has been updated.

If any other command sequence is detected within the Locate Record domain, the nonconforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the storage director determines that there is not enough space on the track to write a record, the operation is terminated with status that includes unit check (Invalid Track Format).

After the last record is written on the track, the rest of the track is erased.

**Programming Note:**

When execution is terminated with status that includes unit check (Invalid Track Format), the residual count in the SCSW<sup>3</sup> will not necessarily indicate the amount by which the record length exceeded the available space on the track.

**Read Tracks (0C):** This Operation code prepares the storage director to transfer all records from one or more consecutively addressed tracks to the channel. Data transfer starts with the first record (including Record Zero) following the orientation specified by the Search Argument and Orientation Modifier parameters. The number of tracks to be operated on is specified by the Count parameter.

Track access is initiated using the Seek Address and Sector Number parameters.

When track access is completed, the search operation specified by the Search Argument and the Orientation Modifiers (byte 0, bits 0-1) is performed.

Locate Record must be followed by the number of Read Track CCWs specified in the Count parameter. If any other command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

**Erase (11):** This Operation code causes the control unit to erase records from one or more consecutively addressed tracks. The number of tracks to be processed is specified by the Count parameter. No data transfer commands may occur in the Locate Record domain. Any command that follows Erase is outside the domain.

If the file mask does not allow Format Write operations (bits 0-1 = '01' or '10'), Locate Record is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Track access is initiated using the Seek Address and Sector Number parameters.

When track access is completed, the search operation specified by the Search Argument and the Orientation modifiers (byte 0, bits 0-1) is performed.

All records on the track, following the record that satisfied the search operation, are erased.

If more tracks remain to be processed after all records on a track have been erased, the control unit attempts to advance to the next track and continue the erase operation starting with the first user data record on the track. If the next track is not within the defined extent, execution is terminated with status that includes unit check (File Protected). If no user data record is found on the next track, execution continues and no exception condition is reported.

Device End status is presented after the erase operation is completed on the last track within the Locate Record domain.

**Read (16):** This Operation code prepares the storage director to transfer one or more consecutive user data records to the channel. The number of records to be read is specified by the Count parameter. Read may also be used to read Home Address and/or Record Zero when Index or Home Address orientation is specified.

If Index or Home Address orientation is specified (byte 0, bits 0-1 = '01' or '11'), a Read Count command may not be suffixed to the Locate Record domain. If a Read Count suffix is specified (Auxiliary byte, bit 7 = '1'), Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

Track access is initiated using the Seek Address and Sector Number parameters. If Index or Home Address orientation is specified, a Sector value of zero is used for track access; the Sector Number parameter is not used.

When track access is completed, the storage director orients the device according to the Orientation modifiers (byte 0, bits 0-1). If Home Address orientation is specified and no Home Address is detected on the track, execution is terminated with status that includes unit check (the sense data is the Data Check appropriate for an 'HA, No Sync Byte' for the device type).

Locate Record must be followed by a sequence of read commands (other than Read Track, Read IPL or Read Multiple Count Key and Data) that is sufficient to process the number of records specified by the count parameter. When Auxiliary byte, bit 7 = '1', the last CCW in the Locate Record domain must be a multi-track Read Count. Command sequence requirements for read commands following Locate Record are as follows:

- All read commands except Read Special Home Address, Read Home Address and Read Record Zero must be multi-track commands.
- When Index orientation is specified, Locate Record may only be followed by a Read Special HA or Read HA CCW.
- When Home Address orientation is specified, Locate Record may only be followed by a Read R0 CCW.
- When Count area orientation is specified, Locate Record may only be followed by a Read Data or Read Key and Data CCW.

- When Data area orientation is specified, Locate Record may only be followed by a Read Count, Read Key and Data, Read Data, or Read Count Key and Data CCW.
- A Read Special HA or Read HA CCW may only be followed by a Read R0 CCW.
- Read R0, Read Count, Read Key and Data, Read Data or Read Count Key and Data commands may be followed by a Read Count, Read Key and Data, Read Data or Read Count Key and Data CCW.

If any other command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If no user data record is detected on the track, execution is terminated with status that includes unit check (No Record Found).

If the operation continues after the last record on a track has been read, the storage director attempts to advance to the next track, as described in “Operations within the domain of a Locate Record” on page 22, and continue the operation with the first user data record on the next track. If the next track does not contain a user data record, execution is terminated with status that includes unit check (No Record Found).

## Locate Record Extended

The Locate Record function is provided by both the Locate Record command and the Locate Record Extended command. See “Locate Record/Locate Record Extended” on page 16.

Locate Record Extended transfers a variable number of parameter bytes, consisting of fixed parameters and extended parameters, from the channel. For ESS the number of parameter bytes transferred is between 20 and 22. The fixed parameters specify the location and number of records or tracks to be processed, the operation to be performed, and other control information. The extended parameters contain additional control information. Execution of Locate Record Extended creates an operational domain as defined in “Locate Record/Locate Record Extended” on page 16.

If Locate Record Extended is received within a Locate Record domain, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If Locate Record Extended was not preceded by a Define Extent or Read IPL command in the same channel program, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). If any other operation is specified, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The format of the parameter list and criteria for validity checking are described in “Locate record extended parameters” on page 42. Parameters and reserved fields are checked for validity in the order that they appear in the parameter list, except that the Operation code (bits 2-7) in the operation byte and the extended operation

code must be validated first. If an exception condition exists in more than one parameter in the list, only the first detected exception is reported.

Channel End status is presented after a valid parameter list is received. Execution of Locate Record Extended proceeds as described for Locate Record (see "Locate Record" on page 29.)

Orientation will be reset in the Storage Director when Locate Record Extended is received. Execution of Locate Record Extended establishes the orientation state specified by the operation byte parameter before any data is transferred to or from the device.

Upon completion of the Locate Record domain, orientation is reset in the Storage Director unless otherwise specified for the specific Locate Record Extended operation.

### **Locate record extended parameters**

The Locate Record Extended parameter list has the following format:

<b>Byte</b>	<b>Name</b>
<b>0</b>	Operation byte
<b>1</b>	Auxiliary byte
<b>2</b>	Must be zero
<b>3</b>	Count
<b>4-7</b>	Seek address
<b>8-12</b>	Search argument
<b>13</b>	Sector number
<b>14-15</b>	Transfer length factor
<b>16</b>	Must be zero
<b>17</b>	Extended operation byte
<b>18-19</b>	Extended parameter length
<b>20-21</b>	Extended Parameter

If bytes 2 and 16 are not zero, Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

**Operation Byte:** Byte 0 specifies the operations to be performed within the Locate Record domain and the orientation to be established at the completion of track access.

Figure 14. Locate Record Extended, Byte 0, Operation Byte

Bits	Value	Definition
0-1		Orientation These bits are orientation modifiers that specify the orientation to be established by the execution of Locate Record Extend when the operation specified in bits 2-7 requires orientation to a specific record or record area before data transfer is initiated.
	00	Count area This orientation specifies that data access for the CCW that immediately follows Locate Record Extended is to begin with the storage director oriented to the count area of the record that satisfied the search operation. That CCW may, therefore, process the key and data areas of the record that satisfied the search operation. Count area orientation is valid with all Operation codes.
	01	Home Address This orientation specifies that data access for the CCW that immediately follows Locate Record Extended is to begin with the storage director oriented to Home Address.  Home Address orientation is valid with the Format Write (03), Read Tracks (0C), Read (16) and Extended (3F) Operation codes only. If the Operation code is X'3F', the extended operation byte must specify Read Trackset (byte 17 = '0E'). If any other operation is specified when Home Address orientation is specified (bits 0-1 = '01'), the Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
	10	Data area This orientation specifies that data access for the CCW that immediately follows Locate Record Extended is to begin with the storage director oriented to the data area of the record that satisfied the search operation. That CCW, therefore, will not process any areas of the record that satisfied the search operation.  Data area orientation is valid with the Extended (3F), Write Data (01), and Read (16) Operation codes only. If the Operation code is X'3F', the extended operation byte must specify Update Write Trackset (byte 17 = '13'). If any other operation code is specified when bits 0-1 = '10', Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
	11	Index This orientation specifies that data access for the CCW that immediately follows Locate Record is to begin with the storage director oriented to Index.  Index orientation is valid with the Format Write (03) and Read (16) operation codes only. If bits 2-7 contain any other operation code when bits 0-1 = '11', the Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
2-7		Operation Code These bits specify the operation to be performed within the Locate Record Extended domain. The Operation codes supported by this storage subsystem are listed below. If any other Operation code is specified, Locate Record is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).  The operations common to both the Locate Record and Locate Record Extended commands are defined in "Locate record operations" on page 34. Operations unique to the Locate Record Extended command are defined in "Locate Record Extended operations" on page 47.
	X'01'	Write Data
	X'03'	Format Write
	X'06'	Read Data
	X'0B'	Write Track
	X'0C'	Read Tracks
	X'11'	Erase
	X'16'	Read
	X'3F'	Extended op code Byte 17 contains an extended operation code to specify the operation within the Locate Record Extended domain.

Figure 15 on page 44 illustrates the valid combinations of orientation and operation code values.

Operation Code	Orientation				Byte 17
	Cnt	HA	Data	Index	
Write Data	01	n	81	n	00
Format Write	03	43	n	C3	00
Write Track	0B	n	n	n	00
Read Tracks	0C	4C	n	n	00
Read	16	56	96	D6	00
Erase	11	n	n	n	00
Write Any	3F	n	n	n	09
Read Any	3F	n	n	n	0A
Read Trackset	3F	7F	n	n	0E

**Note:** n - Combination is not valid.

**Note:** All operations that can be performed in the domain of a Locate Record command, except Orient (X'00') and Read Data (X'06'), can be performed in the domain of a Locate Record Extended command.

**Auxiliary Byte:** Byte 1 is the Auxiliary byte. It is used to validate optional fields in the Locate Record Extended parameter list.

Figure 16. Locate Record Extended, Byte 1, Auxiliary Byte

Bits	Value	Definition
0	0	Bytes 14-15 are unused
	1	Bytes 14-15 contain a Transfer Length Factor that overrides the blocksize specified by the Define Extent parameter.
1-6	0	Must be zero
	1	If any of these bits are '1', the Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).
7	0	No Read Count CCW is suffixed to the Locate Record domain
	1	A Read Count CCW is suffixed to the Locate Record domain.  When this bit is set, it specifies that the last data transfer CCW in the Locate Record domain will be a Read Count CCW. The Read Count CCW is a suffix to the Locate Record domain and the record it operates on is included in the count of records to be processed as specified by the Count parameter.  A Read Count command may only be suffixed to the domain of a Locate Record Extended that specifies a Write Data (01), Write Any (09), Read Any (0A), or Read (16) operation code; if bit 7 = '1' when any other Operation code is specified, Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

**Count:** Byte 3 is the Count parameter. In general, the count parameter specifies the number of records, or tracks to be operated on by data transfer commands that follow Locate Record Extended. Specific interpretation of the Count parameter depends upon the operation code in byte 0 and is defined in "Locate Record Extended operations" on page 47 for each Operation code.

The Count must be nonzero. If Read Count Sufficing is specified in a Locate Record, the count must be greater than 1. If the Count is invalid, Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

**Seek Address:** Bytes 4-7 specify a seek address 'CCHH'.

The value in bytes 4-7 must be a valid primary track address for the device and must be within the extent boundaries specified by the preceding Define Extent command.

If the Seek Address is not valid for the device, Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). If the Seek Address is not within the defined extent, Locate Record Extended is terminated with status that includes unit check (File Protected).

**Search Argument:** Bytes 8-12 specify a value to be used as a search argument for the Locate Record Extended search operation that is described in "Locate record operations" on page 34.

When the operation specified in byte 0 does not require orientation to a specific record, no search operation is performed and bytes 8-12 are ignored. When Home Address orientation is specified, byte 12 is ignored.

**Sector Number:** Byte 13 contains a sector number to which the device is to be positioned before the Storage Director establishes orientation.

The sector number must be within the range of valid sector numbers for the device. If the sector number is invalid, Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

A value of X'FF' is valid and specifies that sector positioning is not to be performed prior to establishing orientation.

**Transfer Length Factor:** When byte 1, bit 0 is '0', bytes 14-15 must contain zeros; if bytes 14-15 are not zero, Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

When byte 1 bit 0 is '1', bytes 14-15 contain a Transfer Length Factor (TLF). The Transfer Length Factor must be non-zero; if it is zero, Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). If the Transfer Length Factor value is greater than the value specified (or implied) in the Define Extent Blocksize parameter, Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

The Storage Director uses the TLF to determine the number of data bytes to be requested from the channel for each write command that follows a Locate Record Extended that specified the Write Data (01) Operation code. The product of the value in bytes 14-15 and the count parameter is used to determine the total number of bytes to be transferred by data transfer commands that are executed within the domain of a Locate Record Extended that specified the Format Write (03), Write Track (0B), or Read (16) Operation codes.

The TLF value is not retained by the Storage Director after the expiration of the Locate Record domain.

If Locate Record Extended does not specify a Transfer Length Factor, the Storage Director will use the value from the Define Extent Blocksize parameter for any required data transfer length calculation.

**Extended Operation Byte:** Byte 17 contains an operation code unique to the Locate Record Extended command. When byte 0, bits 2-7 = X'3F', byte 17 specifies the operation to be performed within the Locate Record domain. Otherwise, the operation specified in byte 0, bits 2-7, is performed and byte 17 must be zero. Byte 17 must be a supported operation code when byte 0, bits 2-7 = X'3F'. If these conditions are not met, the Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

Valid Extended Operation Codes are listed below and described in "Locate Record Extended operations" on page 47.

**Hex Operation Code**

- 09** Write Any
- 0A** Read Any
- 0E** Read Trackset

**Note:** When an operation code common to both Locate Record and Locate Record Extended commands is specified in a Locate Record Extended command it will always be specified in byte 0, bits 2-7. The extended operation byte (byte 17) is used to specify operation codes not supported by the Locate Record command. When these codes are specified, byte 0, bits 2-7 will contain X'3F' to indicate that the operation specified in byte 17 is to be used.

The orientation modifiers are specified in byte 0, bits 0-1 regardless of where the operation code is specified.

**Extended Parameter Length:** Bytes 18-19 contain an unsigned 16-bit binary value that specifies the total number of extended parameter bytes. The format and content of the Extended Parameters are defined by the Extended Operation code.

The length for the Extended Operation codes must be consistent with the Extended Operation code in byte 17 as follows:

<b>Extended Operation Code Length</b>	
<b>09</b>	0001
<b>0A</b>	0001
<b>0E</b>	0001 or 0002

If the operation code is any code other than those defined, the extended parameter length count must be zero. If these conditions are not met the Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

**Extended Parameter:** The Extended Parameter contains data to support the execution of subsequent data transfer commands within this Locate Record domain.



## Locate Record Extended operations

The following subsections describe the processes that occur as a part of the execution of the Locate Record Extended command, when an extended operation code (byte 17) is specified. The operation code (byte 0, bits 2-7) will contain X'3F'.

The processes that occur when an operation code (byte 0, bits 2-7), other than X'3F' is specified, are described in "Locate record operations" on page 34.

**Read Any:** This operation code (X'0A') prepares the Storage Director to read one or more consecutive user data records on a set of tracks without first orienting to a specific record. No search operation is performed. The number of records to be operated upon is specified by the Count parameter.

A set of tracks consists of one or more consecutively addressed tracks within a single cylinder. The Extended Parameter specifies the set size to be used in the Locate Record domain.

The extended parameter length parameter (bytes 18-19) value must equal '1'; otherwise the Locate Record Extended command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

The Extended Parameter must be one byte in length and contains an unsigned 8-bit binary value that specifies the set size to be used in the Locate Record Domain.

The set size must be '1'; otherwise execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). The entire set of tracks as defined by the Extended Parameter must be contained within the defined extent; otherwise execution is terminated with status that includes unit check (File Protected, format X'04', Invalid Parameter).

The Sector Number parameter is not used.

When track access is completed, the Storage Director orients to the beginning of the count area of a user data record and prepares to transfer the associated record. The Search Argument parameter is ignored and no Search operation is performed. If no user data record is found within the track set, the operation is terminated with status that includes unit check (No Record Found).

**Note:** The first record returned when operating in Read Any mode may be any record on the track and has no direct relationship to which record was written first while operating in Write Any mode or any other mode.

When auxiliary byte, bit 7 = '0', Locate Record must be followed by a sequence of multi-track Read Count, Read Count Key and Data, or Read Data commands that is sufficient to process the number of records specified by the count parameter. When auxiliary byte, bit 7 = '1', the last CCW in the Locate Record domain must be a multi-track Read Count command.

If an invalid command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the operation continues after the last record on a track has been read, the Storage Director orients to the beginning of the count area of the first user data record on the next track within the set of tracks and prepares to operate on this record. When the last record on the last track in the set of tracks has been read, the storage director orients to the beginning of the count area of the first user data record on the first track in the set of tracks and prepares to operate on this record.

If the number of read commands in a Locate Record domain exceeds the number of user data records on the track(s), one or more records that were previously read in the Locate Record domain will be re-read, the operation continues, and no exception is reported.

**Programming Note:** The record ID of the first record read may be determined by reading the first count field or by suffixing a Read Count CCW to the Locate Record domain.

1. If the number of records read is equal to the number of user data records on the set of tracks, the count area read by a suffixed Read Count command will be that of the first record that was read.
2. If the number of records read is not equal to the number of user data records on the set of tracks, the count area transferred by the suffixed Read Count command will be that of the record that follows the last record read. The ID of the first record read must be determined by referencing the record ID read, the number of records read, and the known sequence of record IDs on the track(s).

**Write Any:** This operation code (X'09') prepares the Storage Director to update the data area of one or more consecutive user data records on a set of tracks without first orienting to a specific record. No search operation is performed. The number of records to be operated upon is specified by the Count parameter.

A set of tracks consists of one or more consecutively addressed tracks within a single cylinder. The Extended Parameter specifies the set size to be used in the Locate Record domain.

The extended parameter length parameter (bytes 18-19) value must equal '1'; otherwise the Locate Record Extended command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

The Extended Parameter must be one byte in length and contains an unsigned 8-bit binary value that specifies the set size to be used in the Locate Record Domain.

The set size must be '1'; otherwise execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). The entire set of tracks as defined by the Extended Parameter must be contained within the defined extent; otherwise execution is terminated with status that includes unit check (File Protected, format X'04', Invalid Parameter).

**Note:** Extent checking is performed during the validation of Locate Record Extended parameters, not during track access. This will ensure a consistent protection exception in those cases when the set of tracks extends beyond the extent boundary, regardless of whether the track(s) outside the extent were used in the data transfer operation.

If the file mask inhibits all write operations (bits 0-1 = '01'), Locate Record is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The Sector Number parameter is not used.

When track access is completed, the Storage Director orients to the count area of a user data record. The Search Argument parameter is ignored and no Search operation is performed. If no user data record is found within the track set, the operation is terminated with status that includes unit check (No Record Found).

When auxiliary byte, bit 7 = '0', Locate Record Extended must be followed by the number of Write Update Data commands specified by the count parameter; when auxiliary byte, bit 7 = '1', Locate Record Extended must be followed by a number of Write Update Data commands that is one less than the value specified by the Count parameter, and the last Write Update Data command must be followed by a multi-track Read Count command.

If an invalid command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the operation continues after the last record on a track has been updated, the Storage Director orients to the beginning of the count area of the first user data record on the next track within the set of tracks and prepares to operate on this record. When the last record on the last track in the set of tracks has been read, the storage director orients to the beginning of the count area of the first user data record on the first track in the set of tracks and prepares to operate on this record.

The Transfer Length Factor (or the Blocksize from Define Extent, if no Transfer Length Factor was specified) must be equal to the data area length of the records to be updated for operations other than Get Record Set.

If the data area length of any record to be updated differs from the value in the TLF (or Blocksize), execution is terminated with status that includes unit check (Invalid Track Format) and the record is not updated.

If the number of write commands in a Locate Record domain exceeds the number of user data records on the set of tracks, one or more records that were previously updated in the Locate Record domain will be re-written, the operation continues, and no exception is reported.

**Programming note:** The record ID of the first record updated may be determined by suffixing a Read Count command to the Locate Record domain.

1. If the number of records updated is equal to the number of user data records on the set of tracks, the count area read by a suffixed Read Count command will be that of the first record that was updated.
2. If the number of records updated is not equal to the number of user data records on the set of tracks, the count area read by the suffixed Read Count command will be that of the record that follows the last record updated. The ID of the first record updated must be determined by referencing the record ID read, the number of records updated, and the known sequence of record IDs on the track(s).

**Read Trackset - X'0E'**: The Read Trackset Operation Code prepares the Storage Director to transfer all records from one or more tracks to the channel. The tracks to be transferred are specified by the Extended Parameter and the number of tracks to be transferred is specified by the Count Parameter (byte 3).

The maximum length of the Extended Parameter is specified in byte 43 of the Device Characteristics Information (see Appendix A, "Device data, geometries, and facilities" on page 219).

The Extended Parameter contains a bit map that represents a set of sequentially addressed tracks within the defined extent. Each bit in the parameter represent one track. A '1' bit indicates the data associated with the corresponding track is to be read. A '0' bit indicates the track is to be skipped.

The first bit must be a '1' and represents the track whose address is specified in the Seek Address parameter (bytes 4-7). Subsequent bits represent consecutively addressed tracks in ascending order. If the first bit is not a '1', the Locate Record Extended command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

The number of '1' bits in the bit map must be equal to the value in the count parameter (byte 3); otherwise Locate Record Extended is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

All tracks in the bit map represented by the '1' bits must be contained within the defined extent; otherwise the Locate Record Extended command is terminated with status that includes unit check (File Protected).

Track access is initiated using the Seek Address and Sector Number parameters.

When track access is completed, the search operation specified by the Search Argument and the orientation modifiers (byte 0, bits 0-1) is performed.

Locate Record Extended must be followed by the number of Read Track commands specified in the count parameter (byte 3). If any other command sequence is detected within the Locate Record domain, the non-conforming command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

## Seek commands - general

When a Seek, Seek Cylinder, or Seek Head command is received, the applicable elements of the seek address parameter are validated, the directory is searched to determine if the track is in cache and the parameters are saved until data access is required.

If a Set Sector command is received, the sector value is validated and is also saved.

Execution of any seek command resets orientation in the storage director.

## Seek

Seek transfers a six-byte seek address parameter from the channel.

Seek is not valid in the domain of a Locate Record command; if Seek is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the file mask inhibits Seek commands (bits 3-4 are not '00'), execution is terminated with status that includes unit check (File Protected).

The storage director requests a six-byte seek address parameter from the channel; if the channel provides fewer than six bytes, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

The format of the seek address parameter is '00CCHH', where '00' is two bytes of binary zeros and 'CCHH' is a four-byte track address. The track address is checked to ensure that it is a primary track (see Appendix A, "Device data, geometries, and facilities" on page 219). If the seek address parameter is not valid, execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If Seek was preceded in the channel program by a Define Extent command and the seek address is outside the defined extent, execution is terminated with status that includes unit check (File Protected).

After a valid parameter has been received and the cache directory checked to determine if the track is in cache, the track address is saved and Channel End and Device End status is presented.

## Seek Cylinder

Seek Cylinder transfers a six-byte seek address parameter from the channel.

Seek Cylinder is not valid in the domain of a Locate Record command; if Seek Cylinder is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the file mask inhibits Seek Cylinder commands (bit 3 is not '0'), execution is terminated with status that includes unit check (File Protected).

The storage director requests a six-byte seek address parameter from the channel; if the channel provides fewer than six bytes, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

The format of the seek address parameter is '00CCHH', where '00' is two bytes of binary zeros and 'CCHH' is a four-byte track address. The track address is checked to ensure that it is a primary track (see Appendix A, "Device data, geometries, and facilities" on page 219). If the seek address parameter is not valid, execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If Seek Cylinder was preceded in the channel program by a Define Extent command and the seek address is outside the defined extent, execution is terminated with status that includes unit check (File Protected).

After a valid parameter has been received and the cache directory checked to determine if the track is in cache, the track address is saved and Channel End and Device End status is presented.

## Seek Head

Seek Head transfers a six-byte seek address parameter from the channel.

If the Seek Head command has not been preceded in the command chain by a command that establishes track orientation, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Seek Head is not valid in the domain of a Locate Record command; if Seek Head is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the file mask inhibits Seek Head commands (bits 3-4 = '11'), execution is terminated with status that includes unit check (File Protected).

The storage director requests a six-byte parameter from the channel; if the channel provides fewer than six bytes, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

The format of the seek address parameter is '00CCHH', where '00' is two bytes of binary zeros and 'CCHH' is a four-byte track address. The track address is checked to ensure that it is a primary track (see Appendix A, "Device data, geometries, and facilities" on page 219). If the seek address parameter is not valid, execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If Seek Head was preceded in the channel program by a Define Extent command and the seek address is outside the defined extent, execution is terminated with status that includes unit check (File Protected).

The 'HH' part of the received track address replaces the 'HH' value in the current track address.

After a valid parameter has been received and the cache directory checked to determine if the track is in cache, the track address is saved and Channel End and Device End status is presented.

## Set File Mask

Set File Mask causes 1 byte to be transferred from the channel to the storage director.

If Set File Mask is received within the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If Set File Mask was preceded in the same channel program by a Define Extent, another Set File Mask, or a Read IPL, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director attempts to transfer a mask byte from the channel; if the channel does not transfer a data byte, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

The format of the mask byte is identical to the Define Extent Mask byte parameter (see "Define Extent" on page 23). If bit 2 = '1', the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If the mask byte is valid, it replaces the current value of the file mask.

Channel End and Device End status is presented after a valid mask byte parameter has been received.

## Set Sector

Set Sector transfers a one-byte sector number parameter from the channel.

Set Sector is not valid in the domain of a Locate Record command; if Set Sector is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The Set Sector must be preceded in the command chain by a command that specifies (or implies) 'CCHH' parameters. If the read command is not preceded by a Seek, Seek Cylinder, Locate Record, Read IPL, or Recalibrate, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director requests a one-byte sector number from the channel; if the channel does not transfer a data byte, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

The sector number is an unsigned 8-bit binary value that specifies the sector to which the device is to be positioned before execution of the following command is initiated. If the sector number is zero, execution of the following command is to be initiated with the device positioned immediately before Index. A sector value of X'FF' specifies that sector positioning is not to be performed before initiating execution of the following command.

The sector number must be within the range of valid sector numbers for the device; if the sector number is invalid, execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). See Appendix A,

“Device data, geometries, and facilities” on page 219 for the range of valid sector numbers for a specific device.

A sector number of zero causes orientation to index.

If more than one Set Sector command is received and saved before orientation is required, only the last sector number received will be used. Previous values will be discarded.

Any sector value that has been stored and not used at normal or abnormal (unit check) completion of command chaining will be discarded at that time.

**Note:** The Set Sector command may not cause the device to be positioned to a specific record. A search command must be used following Set Sector to verify that the device is positioned to a specific record.

## Read Sector

Read Sector transfers a one-byte sector value to the channel.

If Read Sector is received within the domain of a Locate Record command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

One byte of data is transferred to the channel. The data byte is an unsigned 8-bit binary value that represents the sector position of the count area of the record that was operated on by the search or data transfer command that immediately preceded Read Sector.

In emulation mode, the sector number indicates the sector number on which the record would reside on the emulated device.

If Read Sector was not immediately preceded by a read, write, or search command, a valid but unpredictable sector value is returned to the channel. If the immediately preceding search or data transfer command operated on Home Address or Record Zero, a value of X'00' is transferred to the channel.

Channel End and Device End status is presented at the completion of data transfer.

Execution of Read Sector resets orientation in the storage director.

## Read Device Characteristics

Read Device Characteristics transfers 64 bytes of device characteristics information to the channel.

If Read Device Characteristics is received in the domain of a Locate Record command, it is rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Execution of Read Device Characteristics resets orientation in the storage director.



## Device Characteristics information

The format of the device characteristics information transferred by Read Device Characteristics is described below. The first six bytes are identical to bytes 1-6 of the Sense ID (E4) command (see “Sense ID” on page 87). For specific values of bytes 0-63 see Appendix A, “Device data, geometries, and facilities” on page 219.

Bytes	Contents
0-1	Storage director type
2	Storage director model information
3-4	Device type
5	Device model
6-9	Device and storage director facilities
10	Device class code
11	Device type code
12-13	Number of Primary cylinders
14-15	Tracks per cylinder
16	Number of sectors
17-19	Total track length usable for user data records
20-21	Length of HA and R0
22	Track capacity calculation formula
23-27	Track capacity calculation factors
28-29	Address of first Alternate cylinder
30-31	Number of Alternate tracks
32-33	Address of first Diagnostic cylinder
34-35	Number of Diagnostic tracks
36-37	Address of first Device Support cylinder
38-39	Number of Device Support tracks
40	MDR record ID
41	OBR record ID
42	Storage Director type code
43	Read Trackset Parameter length
44-45	Maximum Record Zero Data length
46	Reserved
47	Track Set Size
48	Additional Track Capacity Calculation Factor
49-50	RPS Sector Calculation Factors
51-53	Reserved
54	Generic Device/CU functions/Features
55	Contain binary zeros
56	Real Control Unit Code
57	Real Device Code
58-63	Reserved

**Bytes 0-1:** contain the four-digit storage director type number, represented as four hexadecimal digits.

**Byte 2:** identifies the storage director model number and indicates the architecture supported by the storage director.

Byte,Bit	Description
2,0-1	Architecture Support
2,2	Non-Gap-Synchronous mode
2,3	Reserved
2,4	Cache information
2,5-7	Control Unit model information

**Bytes 3-4:** contain the four-digit device type number, represented as four hexadecimal digits.

**Byte 5:** identifies the device model number and indicates the presence of device features or device emulation.

**Bytes 6-9:** are used to indicate program-visible facilities implemented in the device and control unit as follows.

Byte, Bit	Description
6,0	Multiple Burst ECC
6,1	Locate Record, Read Tracks (0C)
6,2	Reserved
6,3	Locate Record, Read (16)
6,4-6	Reserved
6,7	Reserved for use by VM (non-full pack minidisk), set to zero by ESS
7,0-7	Reserved
8,0-7	Reserved
9,0	Cache Fast Write
9,1	Multi-Path Lock Facility
9,2	Record Cache
9,3	Track Cache
9,4	Reserved Set to zero
9,5	Fast Write
9,6	Reset Allegiance
9,7	24 Byte Compatibility Sense Format

**Byte 6, bit 0** when '1', indicates that the device uses an error correction code that is capable of correcting multiple error bursts within a record area. Programs that need to determine the location of data checks within the record area will need to execute device specific channel commands to acquire this data.

**Byte 6, bit 1** when '1', indicates that the device supports the Read Tracks operation code (0C) in Locate Record command and executes the Read Track command (DE) as defined in "Locate record operations" on page 34.

**Byte 6, bit 3** when '1', indicates that the device supports the Read (16) operation code in the Locate Record command and that read commands have been implemented as defined in "Locate record operations" on page 34.

**Byte 9, bit 0** when '1', indicates that the subsystem supports Cache Fast Write.

**Byte 9, bit 1** Multi-Path Lock Facility set to '1', indicates that the subsystem supports the Multi-Path Lock Facility. This facility provides named locks that software can use to manage multi-host access to shared data.

The Multi-Path Lock Facility is supported in the following modes:

When the ESS operates in either Basic Operations Mode or Enhanced Operations Mode 1, Multi-Path Lock Facility operates as described in this book. In addition to channel command descriptions, see Chapter 4, "Multi-Path Lock Facility" on page 155.

When ESS operates in TPF Mode (TPF Mode is set in VPD), record cache with Multi-Path Lock Facility (RPQ 8B0178) operates as described in the *TPF Support RPQs, GA32-0134*.

**Byte 9, bit 2** Record Cache when '1', indicates that the subsystem supports a record cache. In this type of cache, the unit of data stored in the cache and recorded in the cache directory is a single record.

**Byte 9, bit 3** when '1', indicates that the subsystem supports a track cache. In this type of cache, the unit of data stored in the cache and recorded in the cache directory is a DASD device track image.

**Byte 9, bit 4** Reserved

**Byte 9, bit 5** when '1', indicates that the subsystem supports the DASD Fast Write facility.

**Byte 9, bit 6** when '1', indicates that the subsystem supports the Reset Allegiance command.

**Byte 9, bit 7** when '1', indicates that the subsystem transfers 32 bytes of sense data in the 24 Byte Compatibility format.

Unassigned bits are set to zero.

**Byte 10:** contains an 8-bit device class code. Device class codes for each device are assigned by the device support programming organization.

**Byte 11:** contains an 8-bit device type code. Device type codes for each device are assigned by the device support programming organization.

**Bytes 12-13:** contain an unsigned, 16-bit binary value that equals the number of Primary cylinders on the device. The highest Primary cylinder address on the device is one less than the value in bytes 12-13.

**Bytes 14-15:** contain an unsigned, 16-bit binary value that equals the number of tracks in a cylinder. The highest valid track address in any cylinder is one less than the value in bytes 14-15.

**Byte 16:** contains an unsigned, 8-bit binary value that equals the number of sectors in a track. The highest valid sector number for the device (excluding X'FF') is one less than the value in byte 16.

**Bytes 17-19:** contain an unsigned, 24-bit binary value equal to the total usable capacity of each track, in bytes, after the track has been formatted with a Home Address and a standard Record Zero (key length equals zero and data length equals eight). This is the maximum number of bytes available for formatting user data records, and includes all space that will be occupied by count areas and device-dependent recording items (that is, gaps, ECC bytes, and so on) associated with the records.

**Note:** A set of user data records can be written on a track if the sum of the space values of the records, calculated by the formula specified by byte 22, is equal to or less than the value contained in bytes 17-19.

**Bytes 20-21:** contain an unsigned 16-bit binary value that equals the total amount of space, in bytes, occupied by a Home Address and a standard Record Zero (key length equals zero and data length equals eight). This value includes the space occupied by device-dependent recording items (that is, gaps, ECC bytes, and so on) associated with Home Address and Record Zero.

**Byte 22:** contains an 8-bit unsigned binary value that specifies the formula that is to be used to perform track capacity calculations for the device.

A value of 1 specifies that the amount of space occupied by a record is given by the formula:

$$\text{Space} = (\text{FL1} + \text{FL2})$$

Where:

- Each FL1 and FL2 are rounded up to an integral multiple of F1.
- $\text{FL1} = (\text{F2} + \text{DL})$
- $\text{FL2} = (\text{F3} + \text{KL})$  (FL2 is set to zero when the key length is zero.)

Where:

- F1 is an 8-bit unsigned binary value in byte 23
- F2 is a 16-bit unsigned binary value in byte 24-25
- F3 is a 18-bit unsigned binary value in byte 26-27

A value of 2 specifies that the amount of space occupied by a record is given by the formula:

$$\text{Space} = (\text{FL1} + \text{FL2})$$

Where:

- FL1 and FL2 are rounded up to an integral multiple of F1.
- $\text{FL1} = (\text{F1} \times \text{F2}) + (\text{DL} + \text{F6} + (\text{F4} \times \text{INT1}))$
- $\text{FL2} = (\text{F1} \times \text{F3}) + (\text{KL} + \text{F6} + (\text{F4} \times \text{INT2}))$  (FL2 is set to zero when the key length is zero.)
- $\text{INT1} = \text{Smallest integer} \geq ((\text{DL} + \text{F6}) / (2 \times \text{F5}))$
- $\text{INT2} = \text{Smallest integer} \geq ((\text{KL} + \text{F6}) / (2 \times \text{F5}))$
- F1 is an 8-bit unsigned binary value in byte 23
- F2 is an 8-bit unsigned binary value in byte 24
- F3 is an 8-bit unsigned binary value in byte 25
- F4 is an 8-bit unsigned binary value in byte 26
- F5 is an 8-bit unsigned binary value in byte 27
- F6 is an 8-bit unsigned binary value in byte 48

**Bytes 23-27:** contain factors to be used in the track capacity calculation formulas. See preceding Byte 22 descriptions for format information.

**Bytes 28-29:** contain an unsigned, 16-bit binary value that is the address 'CC' of the first cylinder that contains Alternate tracks.

**Bytes 30-31:** contain an unsigned, 16-bit binary value equal to the number of Alternate tracks implemented on the device.

**Bytes 32-33:** contain an unsigned, 16-bit binary value that is the address 'CC' of the first cylinder that contains Diagnostic tracks.

**Bytes 34-35:** contain an unsigned, 16-bit binary value equal to the number of Diagnostic tracks implemented on the device.

**Bytes 36-37:** contain an unsigned, 16-bit binary value that is the address 'CC' of the first cylinder that contains Device Support tracks.

**Bytes 38-39:** contain an unsigned, 16-bit binary value equal to the number of Device Support tracks implemented on the device.

**Byte 40:** contains an 8-bit MDR record ID for the device. MDR record IDs are assigned by the device support programming organization.

**Byte 41:** contains an 8-bit OBR record ID for the device. OBR record IDs are assigned by the device support programming organization.

**Byte 42:** contains an 8-bit Storage Director Type code that identifies the Storage Director Type, model, and features. Storage Director type codes are assigned by the device support programming organization.

**Byte 43:** contains an unsigned, 8-bit binary value that is the maximum length in bytes this device supports for the extended parameter of the Read Trackset extended operation in the Locate Record Extended command.

**Bytes 44-45:** contain an unsigned, 16-bit binary value that equals the maximum data area length for a Record Zero when the key length equals zero. When the device is emulating some other device type, this value reflects the native device track format.

**Byte 47:** contains an unsigned, 8-bit binary value that defines the roll mode track set size.

**Byte 48:** contains a factor used in the track capacity calculation formulas. The format and content of this byte is uniquely defined for each track capacity formula that may be specified in byte 22.

When byte 22 contains a value of 1, byte 48 is unassigned and will contain binary zeros.

When byte 22 contains a value of 2, byte 48 contains the 8-bit unsigned binary value of F6. F6 is the number of block check and CRC bytes per area.

**Bytes 49-50:** contain factors to be used in the RPS Sector calculation formula.

- Byte 49 contains the 8-bit unsigned binary value of 'F7' that is one-half of the sum of:
  - the number of bytes between the end of Record Zero and the cell containing the start of Record 1, plus
  - the difference in length, in bytes, between a standard RPS sector and RPS sector 0.
- Byte 50 contains the 8-bit unsigned binary value of 'F8' that is the number of cells per RPS sector.

The appropriate sector number for a Locate Record or Set Sector command is smallest integer equal to or less than the result of the following formula:

$$\text{Sector} = (\text{C0} + (2 \times \text{F7}) + \text{NetSpace}) / (\text{F1} \times \text{F8})$$

Where:

- C0 is an unsigned 16-bit binary number contained in bytes 20-21.
- F7 is an unsigned 8-bit binary number contained in byte 49.
- NetSpace is the sum of the Space values for all records on the track prior to the target record, other than Record 0, calculated by the formula specified by byte 22.
- F1 is an unsigned 8-bit binary number contained in byte 23.
- F8 is an unsigned 8-bit binary number contained in byte 50.

**Byte 54:** Identifies Device and control unit features.

Byte, Bit	Description
54,0	Reserved
54,1	RAID device (set to one for ESS)
54,2-3	Reserved
54,4	Split CE/DE (set to one when ESS splits CE/DE, otherwise, set to zero)
54,5-7	Reserved

**Byte 56:** real CU type code.

When this byte is non-zero, byte 42 contains the emulated CU type and this byte contains the real CU type.

**Byte 57:** real device type code.

When this byte is non-zero, byte 41 contains the emulated device OBR code and this byte contains the real OBR code.

---

## Read commands

The read commands transfer recorded data from the buffer, cache or the device to the channel. The storage director orients to a specific record area and transfers one or more areas or records to the channel.

All read commands except Read IPL must be preceded in the command chain by a command that specifies (or implies) 'CCHH' parameters. If the read command is not preceded by a Seek, Seek Cylinder, Prefix, Locate Record, or Read IPL the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

All read commands, with the exception of Read Track, are valid outside the domain of a Locate Record.

The Read Multiple CKD and Read IPL commands are not valid inside the domain of a Locate Record.

If a Define Extent command is received following the Seek or Seek Cylinder and the read command results in orientation with the device or cache, the Storage Director determines whether the track to be accessed is within the extent limits communicated by a previous Define Extent command. If the track is not within the

specified boundaries, the read command is rejected with unit check (File Protected) included in ending status.

Most read commands may be executed in multi-track mode (see “Operations outside the domain of a Locate Record” on page 23).

If the channel truncates data transfer before all data bytes have been transferred, the storage director continues processing data until the end of the record is reached. No exception condition is reported as the result of the truncation of data transfer by the channel.

Channel End and Device End status is presented after the storage director completes processing of the record. If an exception condition is detected after the channel has terminated data transfer, Unit Check status is included with Channel End and Device End.

Of the eleven read commands available to ESS three commands, Read Special Home Address, Read Multiple CKD, and Read IPL are only defined as single-track mode commands. Read Track is defined as a multi-track command. All other commands may be executed in either mode.

See Figure 5 on page 19 for a complete list of channel commands.

## Read Special Home Address

Read Special Home Address transfers a complete Home Address to the channel.

If the Read Special Home Address command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read Special Home Address is valid in the domain of a Locate Record that specified a Read (16) Operation code with Index Orientation (byte 0, bits 0-1 = '11'). If it is received within any other Locate Record domain, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

When received outside the domain of a Locate Record, Read Special Home Address is valid only if the channel program is executing with Diagnostic access authorization. If the file mask does not specify Diagnostic access authorization (bits 5-6 not = '10'), execution is terminated with status that includes unit check (format X'05', Diagnostic Command Violates File Mask).

If no Home Address is detected on the track, execution is terminated with status that includes unit check (the sense data is the Data Check appropriate for an 'HA, No Sync Byte' for the device type).

Channel End and Device End status is presented at the completion of data transfer.

At the completion of command execution, the storage director is oriented to Home Address.

## Read Home Address

Read Home Address transfers a five-byte Home Address to the channel.

If the Read Home Address command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read HA is valid in the domain of a Locate Record that specified a Read Data (06) Operation code, or a Read (16) Operation code and Index Orientation (byte 0, bits 0-1 = '11'). If it is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

When received outside the domain of a Locate Record, if the multi-track bit (bit 0) in the CCW operation code is '0', the storage director orients to Index and transfers the Home Address to the channel. If the multi-track bit is '1', the storage director orients to Index, attempts to advance to the next track as described in "Operations outside the domain of a Locate Record" on page 23, and transfers the Home Address of that track to the channel.

Channel End and Device End status is presented at the completion of data transfer.

At the completion of command execution, the storage director is oriented to Home Address.

**Programming Note:** When received outside the domain of a Locate Record, Read Home Address orients to Index prior to transferring data; therefore, a multi-track Read HA will always switch to the next track before transferring data.

## Read Record Zero

Read Record Zero transfers an entire Record Zero to the channel.

If the Read Record Zero command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read R0 is valid in the domain of a Locate Record that specified a Read Data (06) Operation code, or a Read (16) Operation code and Index or Home Address orientation (byte 0, bits 0-1 = '11' or '01'). If it is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Within the domain of a Locate Record, the storage director will be oriented to Home Address when execution of Read R0 is initiated. Data transfer begins with the following R0 count area.

When received outside the domain of a Locate Record, if the storage director is oriented to Home Address when Read R0 is received, data transfer begins with the following R0 count area; otherwise, the storage director orients to Index, and



execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the beginning of the Record Zero count area on the same track and transfers the record to the channel.
- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations outside the domain of a Locate Record" on page 23, orients to the beginning of the Record Zero count area, and transfers the record to the channel.

If there is no Record Zero on the track, execution is terminated with status that includes unit check (No Record Found).

Channel End and Device End status is presented at the completion of data transfer.

The key length of R0 is always zero, therefore no key area transfer is attempted.

At the completion of command execution, the storage director is oriented to the Record Zero data area.

**Programming Note:** When executing outside the domain of a Locate Record, if the storage director is not oriented to HA when execution of a multi-track Read R0 is initiated, the command will always advance to the next track before transferring Record Zero.

## Read Count Key and Data

Read Count Key and Data transfers an entire record, other than Record Zero, to the channel.

If the Read Count, Key, and Data command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, Locate Record Extended, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read CKD is valid in the domain of a Locate Record that specified a Read (16) or Read Data (06) Operation code or in the domain of a Locate Record Extended that specified a Read Any (0A) Operation code. If Read Count, Key, and Data is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director orients to the beginning of the next count area and transfers the count, key, and data areas of that record.

If End-of-Track is detected before a count area is found, when executing within the domain of a Locate Record, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the beginning of the count field of the first user data record on the same track and transfers the count, key and data areas of the record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).

- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations within the domain of a Locate Record" on page 22, and transfers the count key and data areas of the first user data record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).

If End-of-Track is detected before a count area is found when executing outside the domain of a Locate Record, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the beginning of the count field of the first user data record on the same track and transfers the count, key and data areas of the record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations outside the domain of a Locate Record" on page 23.

Channel End and Device End status is presented at the completion of data transfer.

If the key length field in the count area contains zero, no key area transfer is attempted.

If the data length field in the count area contains zero, no data area transfer is attempted and execution is terminated with Channel End, Device End, and Unit Exception status.

At the completion of command execution, the storage director is oriented to the data area of the record that was read.

## Read Key and Data

Read Key and Data transfers the key and data areas of a record to the channel.

If the Read Key and Data command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read KD is valid in the domain of a Locate Record that specified a Read (16) or Read Data (06) Operation code. If it is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the storage director is oriented to a count area when command execution is initiated, it transfers the key and data areas of the same record; otherwise, it orients to the next count area (other than Record Zero) and transfers the key and data areas of that record.

If End-of-Track is detected before a count area is found, when executing within the domain of a Locate Record, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the count field of the first user data record on the same track and transfers the key and data areas of the record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations within the domain of a Locate Record" on page 22, and transfers the key and data areas of the first user data record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).

If End-of-Track is detected before a count area is found when executing outside the domain of a Locate Record, according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the count area of the first user data record on the same track and transfers the key and data areas of that record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations outside the domain of a Locate Record" on page 23, and continues until either a user data record is found, or the operation is terminated with an End-of-Cylinder exception.

Channel End and Device End status is presented at the completion of data transfer.

If the key length field in the count area contains zero, no key area transfer is attempted.

If the data length field in the count area contains zero, no data area transfer is attempted and execution is terminated with Channel End, Device End, and Unit Exception status.

At the completion of command execution, the storage director is oriented to the data area of the record that was read.

## Read Data

Read Data transfers the data area of a record to the channel.

If the Read Data command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, Locate Record Extended, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read Data is valid in the domain of a Locate Record that specified a Read (16) or Read Data (06) Operation code or in the domain of a Locate Record Extended that specified a Read Any (0A) Operation code. If Read Data is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the storage director is oriented to a count or key area when command execution is initiated, it transfers the data area of the same record; otherwise, it orients to the beginning of the next count area (other than Record Zero) and transfers the data area of that record.

If End-of-Track is detected before a count area is found when executing within the domain of a Locate Record, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the count field of the first user data record on the same track and transfers the data area of the record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations within the domain of a Locate Record" on page 22, and transfers the data area of the first user data record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).

If End-of-Track is detected before a count area is found when executing outside the domain of a Locate Record, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the count area of the first user data record on the same track and transfers the data area of that record. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations outside the domain of a Locate Record" on page 23, and continues until either a user data record is found, or the operation is terminated with an End-of-Cylinder exception.

Channel End and Device End status is presented at the completion of data transfer.

If the data length field in the count area contains zero, no data area transfer is attempted and execution is terminated with Channel End, Device End, and Unit Exception status.

At the completion of command execution, the storage director is oriented to the data area of the record that was read.

## Read Count

Read Count transfers the count area of a record, other than Record Zero, to the channel.

If the Read Count command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, Locate Record Extended, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read Count is valid in the domain of a Locate Record that specified a Read (16) or Read Data (06) Operation code or in the domain of a Locate Record Extended that specified a Read Any (0A) Operation code. Read Count is also valid when suffixed to the domain of a Locate Record that specified a Write Data (01) Operation code and suffixed Read Count CCW (auxiliary byte, bit 7 equals '1'). If Read Count is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director orients to the beginning of the next count area and transfers that count area to the channel.

If End-of-Track is detected before a count area is found when executing within the domain of a Locate Record, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the beginning of the count field of the first user data record on the same track and transfers the count area of the record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations within the domain of a Locate Record" on page 22, and transfers the count area of the first user data record to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).

If End-of-Track is detected before a count area is found when executing outside the domain of a Locate Record, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the beginning of the count area of the first user data record on the same track and transfers the count area to the channel. If there is no user data record on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track, as described in "Operations outside the domain of a Locate Record" on page 23, and continues until either a user data record is found, or the operation is terminated with an End-of-Cylinder exception.

Channel End and Device End status is presented at the completion of data transfer.

At the completion of command execution, the storage director is oriented to the count area that was transferred.

## Read Multiple Count Key and Data

Read Multiple Count Key and Data transfers one or more complete records from a device track to the channel.

If the Read Multiple CKD command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read Multiple CKD is not valid in the domain of a Locate Record command; if Read Multiple CKD is received in a Locate Record domain, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director orients to the beginning of the count area of the next user data record and transfers each count, key, and data area on the track until End-of-Track is reached. If End-of-Track is detected before a count area is found, no data is

transferred and Channel End and Device End status is presented; no exception condition is reported.

Channel End and Device End status is presented at the completion of data transfer.

If a zero length data area is encountered, no data field transfer is attempted, no exception status is presented, and the read operation continues with the next count area (if any) on the track.

Read Multiple CKD does not operate upon a track that is marked defective (Home Address Flag byte, bits 6-7 = '10'). If Read Multiple CKD accesses a defective track, the operation is performed on the assigned alternate track; if no alternate track has been assigned, or if an invalid alternate track has been assigned, execution is terminated with status that includes unit check (Command Reject - format X'0B', Invalid Alternate Track Pointer).

If the channel truncates data transfer due to exhaustion of the CCW byte count, the storage director continues processing data from the track until the end of the current track is reached and then presents Channel End and Device End status. No exception condition is reported as the result of the channel's truncation of data transfer. If an exception condition is detected after the channel has truncated data transfer, Unit Check status is included with Channel End and Device End.

If Read Multiple CKD is followed by a search or read command (other than RMCKD), the storage director orients to Index before beginning execution of the next command. If the RMCKD is followed by another RMCKD, the operation of the second RMCKD is unpredictable.

**Note:** The Read Multiple CKD CCW is intended for use by dump/restore programs. The ability to start the transfer at a point other than the beginning of the track is provided for ERP restart.

The byte count in the Read Multiple CKD CCW should contain a value that is at least as large as the length of the maximum size record that can be written on the track.

## Read Track

Read Track transfers multiple records from a track to the channel.

Read Track is valid only in the domain of a Locate Record command that specified the Read Tracks operation code or a Locate Record Extended command that specifies Read Track, or a Read Trackset operation code. If Read Track is received outside the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If Read Track was not immediately preceded by a Locate Record or a Locate Record Extended command with an operation that allows Read Track or another Read Track command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The first Read Track command in a Locate Record or Locate Record Extended domain orients to the beginning of the first count area (including R0) following the orientation established by the Locate Record command and transfers each count, key, and data area on the track until End-of-Track is reached. If End-of-Track is

detected before a count area other than Record Zero is found, only a pseudo count area consisting of eight bytes of X'FF' is transferred and no exception condition is reported.

The second and subsequent Read Track commands in a domain attempt to advance to the next track, orient to Home Address, and transfer each count, key, and data area (including R0) on the track until End-of-Track is reached. If no Home Address is detected on the next track, execution is terminated with status that includes unit check (the sense data is the Data Check appropriate for an 'HA, No Sync Byte' for the device type).

**Note:** If the track is in cache as the result of a Predictable Write and is accessed with Home Address orientation by a 'Read Trackset' operation with Cache Fast Write in effect, a pseudo standard Record Zero will be transferred rather than getting R0 from the device.

After the last record on the track has been transferred, a pseudo count area consisting of eight bytes with a value of X'FF' is transferred to the channel.

Channel End and Device End status is presented at the completion of data transfer.

If a zero length data area is encountered, no data area transfer is attempted, no exception status is presented, and the read operation continues with the next count area, if any, on the track.

Read Track will not operate upon a track that is marked defective (Home Address Flag byte, bits 6-7 = '10'). If Read Track accesses a defective track, the operation execution is terminated with status that includes unit check (Command Reject - format X'0B', Invalid Alternate Track Pointer).

If the channel truncates data transfer due to exhaustion of the CCW byte count, the storage director continues processing data from the device until the end of the track is reached, and then presents Channel End and Device End status. No exception condition is reported as the result of the channel's truncation of data transfer. If an exception condition is detected after the channel has truncated data transfer, Unit Check status is included with Channel End and Device End.

At the completion of command execution the control unit is oriented to the data area of the last record on the track.

**Note:** The Read Track CCW is primarily intended to enable a program to transfer all records from a device track without first having to determine the number of records on the track and their formats.

The byte count in the Read Track CCW should contain a value that is at least as large as the length of the maximum size record that can be written on the track, plus the length of the R0 count and data areas, plus eight bytes for the pseudo count area.

If channel data transfer is truncated due to an exhausted byte count or detection of an exception condition, no pseudo count area will be transferred to indicate the end of the track image character string in the host system storage.

## Read IPL

Read IPL transfers the data field of the first record following Record Zero on Primary Track 0 (that is, cylinder 0 head 0).

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

A Read IPL may not be preceded in the same channel program by a Define Extent or Set File Mask command. If these conditions are violated, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Read IPL executes as though it were a Read Data (06) command that was preceded by Define Extent and Locate Record commands with the following parameters:

Define Extent Parameters:

- Mask Byte = X'00'
- Global Attributes = X'C0'
- Blocksize = 0
- Beginning of Extent Address = 0
- End of Extent Address = highest Primary Track address on the device.

Locate Record Parameters:

- Operation Byte = X'86'
- Auxiliary Byte = X'00'
- Count = 2
- Seek Address = 0
- Search Argument = 0
- Sector Number = 0
- TLF = 0

Channel End and Device End status is presented when data transfer is completed.

**Programming Note:** Read IPL will transfer the data field of the first record following Record Zero on track '0000'. The data read is interpreted as an IPL PSW and one or two CCWs. The first CCW should be a Read CCW that will read the second record on track '0000'. If the IPL sequence must read or write additional records, the second CCW must be a TIC to a command chain read into storage by the preceding Read CCW.

---

## Search commands

The search commands are used to orient to a specific record area on the track before executing a read or write command. The storage director transfers a search argument from the channel, reads a record area from the device or cache and performs a logical compare operation of the two.

All search commands must be preceded in the command chain by a command that specifies (or implies) 'CCHH' parameters. If the search command is not preceded by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated



with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Except when Special Search Key Mode is active (bits 0-1 of Define Extent byte 7 = '10'b), each execution of a search command operates on only one record. The storage director indicates a successful compare by including Status Modifier in ending status.

Search commands are not valid in the domain of a Locate Record command; if a search command is received within the domain of a Locate Record command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

All search commands may be executed in multi-track mode (see "Operations outside the domain of a Locate Record" on page 23).

**Programming Note:** When using multi-track search CCWs, it is important to start the search at the beginning of the track (for example, by using a Read Home Address or Read Record Zero command). If the search is not begun at the start of the track, it is possible that the target area may have passed the read mechanism before the first search operation is initiated; if so, a successful compare will not occur since detection of End-of-Track causes the search operation to continue on the *next track*.

See Figure 5 on page 19 for a complete list of channel commands.

## Search Home Address Equal

Search Home Address Equal transfers a four-byte search argument from the channel and compares it with the track address field in the Record Zero count field of the track image.

If this command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Search HA Equal is not valid in the domain of a Locate Record command; if Search HA Equal is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the multi-track bit (bit 0) in the CCW operation code is '0', the storage director orients to Index and prepares to operate on the Home Address on the accessed track.

If the multi-track bit is '1', the storage director orients to Index, attempts to advance to the next track as described in "Operations outside the domain of a Locate Record" on page 23, and prepares to operate on the Home Address on that track.

The storage director requests a four-byte search argument from the channel; if the channel transfers fewer than four bytes, the data received is compared with an equal number of bytes from the Home Address.

Channel End and Device End status is presented after the compare operation is complete. If the search argument compared equal to the corresponding bytes in the Home Address, Status Modifier is included in ending status.

If no Home Address is detected on the track, execution is terminated with status that includes unit check (the sense data is the Data Check appropriate for an 'HA, No Sync Byte' for the device type).

At the completion of command execution, the storage director is oriented to the Home Address that was operated upon.

## Search ID Equal

Search ID Equal transfers a five-byte search argument from the channel and compares it with the record ID in a count area of the track image.

If this command is not preceded in the command chain by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Search ID Equal is not valid in the domain of a Locate Record command; if Search ID Equal is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the storage director is oriented to Home Address when the command is received, it operates on the Record Zero count area; otherwise, the storage director orients to the beginning of the next count area and operates on that area. If End-of-Track is detected before a count area is found, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to Home Address and prepares to operate on the Record Zero count area on the same track. If no Home Address is detected on the track, execution is terminated with status that includes unit check (the sense data is the Data Check appropriate for an 'HA, No Sync Byte' for the device type). If there is no R0 on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track as described in "Operations outside the domain of a Locate Record" on page 23, and continues until a Record Zero count area is found, or until the operation is terminated with an End-of-Cylinder exception.

The storage director requests a five-byte search argument from the channel; if the channel transfers fewer than five bytes, the data received is compared with an equal number of bytes of the record ID.

Channel End and Device End status is presented after the compare operation is complete. If the search argument compared equal to the corresponding bytes in the record ID, Status Modifier is included in ending status.

At the completion of command execution, the storage director is oriented to the count area that was operated upon.

## Search ID High

Search ID High operates exactly like Search ID Equal except that Status Modifier is included in ending status when the record ID compared high to the search argument.

## Search ID Equal or High

Search ID Equal or High operates exactly like Search ID Equal except that Status Modifier is included in ending status when the record ID compared equal or high to the search argument.

## Search Key Equal

Search Key Equal transfers a search argument from the channel and compares it with a key area of the track image.

If this command is not preceded by a Seek, Seek Cylinder, Locate Record, or Read IPL, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Search Key Equal is not valid in the domain of a Locate Record command; if Search Key Equal is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the storage director is oriented to a count area when the command is received, it operates on the key area of the same record; otherwise, the storage director orients to the count area of the next user data record on the track and operates on the key area of that record. If End-of-Track is detected before a count area is found, execution continues according to the state of the multi-track bit (bit 0) in the CCW operation code:

- If the multi-track bit is '0', the storage director orients to the count area of the first user data record on the track. If no user data record is found on the track, execution is terminated with status that includes unit check (No Record Found).
- If the multi-track bit is '1', the storage director attempts to advance to the next track as described in "Operations outside the domain of a Locate Record" on page 23, and continues until a count area (other than Record Zero) is found, or until the operation is terminated with an End-of-Cylinder exception.

The length of the key area (KL) is defined in the key length field of the count area. If KL is zero, no data is transferred from the channel and no compare operation is performed; the storage director orients to the data area of the record and presents Channel End and Device End status.

The storage director requests a KL-byte search argument from the channel; if the channel transfers fewer than KL bytes, the search argument is compared with an equal number of bytes from the key area.

Channel End and Device End status is presented after the compare operation is complete. If the search argument compared equal to the corresponding bytes in the key area, Status Modifier is included in ending status.

Following execution of a Search Key Equal command, the storage director is oriented to the key area that was operated upon; if the key length was zero, the storage director is oriented to the data area.

**Programming Note:** A Read Data command that immediately follows a Search Key CCW that operated on a record with a key length of zero, will transfer the data area of the first record that follows the record operated on by the search command.

## Search Key High

Search Key High operates exactly like Search Key Equal except that Status Modifier is included in ending status when the key area compared high to the search argument.

## Search Key Equal or High

Search Key Equal or High operates exactly like Search Key Equal except that Status Modifier is included in ending status when the key area compared equal or high to the search argument.

---

## Write commands

The write commands transfer data from the channel and format new records or update areas of existing records.

Write commands are divided into two sub-classes, formatting write commands and update write commands.

Formatting write commands establish a count area on the track, and format the key and data areas of a record. These commands, if not followed immediately by another formatting write command, always cause the remainder of the track to be erased after writing the last area of a record. The formatting write commands are:

- Write Special Home Address
- Write Home Address
- Write Record Zero
- Write Count, Key, and Data
- Write CKD Next Track
- Erase

Formatting write commands must be used to change the format of a record, that is, the length of the key or data areas of a record.

Update write commands update the contents of the key and data areas of an existing record without affecting other areas of that record or other records on the same track. The update write commands are:

- Write Key and Data
- Write Data
- Write Update Key and Data
- Write Update Data

Write commands, except Write Special Home Address, that are executed outside a Locate Record domain must be preceded by a search command that compared

equal on all bytes of the field searched. The Orient Operation code of Locate Record does not satisfy the search prerequisite for a write command.

See Figure 5 on page 19 for a complete list of channel commands.

## Write Special Home Address

Write Special Home Address transfers data from the channel and formats a new Home Address on the track. The data transferred from the channel includes all device dependent data that is recorded with the Home Address. The number of bytes to be transferred, and their format, is device dependent.

If this command is not preceded by a Seek, Seek Cylinder, Locate Record, Read IPL, or Recalibrate, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Write Special Home Address is valid in the domain of a Locate Record command that specified a Format Write (03) Operation code and Index orientation (byte 0, bits 0-1 = '11'). If the command is received within a Locate that does not meet these conditions, it is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The command is valid outside the domain of a Locate Record only if the channel program is executing with Diagnostic access authorization and allows writing HA. If the file mask does not specify 'allow all writes' (bits 0-1 not = '11'), execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). If the file mask does not specify Diagnostic access authorization (bits 5-6 not = '10'), execution is terminated with status that includes unit check (format X'05', Diagnostic Command Violates File Mask).

The storage director attempts to transfer a complete Home Address from the channel; if the channel provides fewer bytes than are requested, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required). The existing HA, R0, and user data records on the track are not affected.

The Home Address data received from the channel are checked for validity. Validity criteria is listed below.

- The flag byte must be zero.
- The Physical Address must equal the address of the currently accessed track.
- The Track address must equal the address of the currently accessed track and the record ID must be zero.

If the Home Address data are not valid, execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). The existing HA, R0, and user data records on the track are not affected.

Channel End and Device End status is presented when the write operation is complete at the device. If Write Home Address is not followed by a Write R0 command, the rest of the track is erased after R0 is written.

At the completion of command execution, the storage director is oriented to Home Address.

## Write Home Address

Write Home Address transfers data from the channel and reformats the Home Address on the track.

Write Home Address is valid in the domain of a Locate Record that specified a Format Write (03) Operation code and Home Address orientation (byte 0, bits 0-1 = '01'). If it is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

When received outside the domain of a Locate Record, Write Home Address must be immediately preceded by a Search Home Address command that compared equal on all four bytes of the track address field in the Home Address. If Write Home Address is preceded by any other command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). If the file mask does not permit writing HA (bits 0-1 not = '11'), execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director requests a five-byte Home Address from the channel; if the channel provides fewer than five bytes, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required). The existing R0, and user data records on the track are not affected.

The Home Address data received from the channel are checked for validity as follows:

- The track address must equal the address of the currently accessed track.
- The flag byte must be zero.

If the Home Address data are not valid, execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). The existing R0, and user data records on the track are not affected.

Channel End and Device End status is presented when the write operation is complete at the device.

At the completion of command execution, the storage director is oriented to Home Address.

## Write Record Zero

Write Record Zero transfers data from the channel and formats a new Record Zero on the track. The first eight bytes transferred are the count area; the remaining bytes are written into the key and data areas.

Write Record Zero is valid in the domain of a Locate Record that specified a Format Write (03) Operation code and Home Address or Index orientation (byte 0, bits 0-1 = '01' or '11'). If it is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

When received outside the domain of a Locate Record, Write Record Zero must be preceded by either:

- A Search Home Address command that compared equal on all four bytes of the Home Address, or
- A Write Home Address command that executed outside the domain of a Locate Record.

If Write R0 is preceded by any other command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). If the file mask does not permit writing R0 (bits 0-1 not = '11'), execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director requests an eight-byte count area from the channel. If the channel provides fewer than eight bytes, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

ESS checks the count field to ensure that:

- The first four bytes (the Track address) must equal the address of the currently accessed track.
- The fifth byte (the record ID) must be zero.
- The sixth byte (the key length) must be zero.
- The seventh and eighth bytes (the data length) must be eight).

If these conditions are not met the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

The storage director prepares to transfer the data area from the channel. If the channel provides fewer than eight bytes, binary zeros are generated to fill the remaining bytes of the data area.

Channel End and Device End status is presented when the write operation is complete at the device. If Write Record Zero is not followed by a Write Count, Key, and Data command, the remainder of the track is erased after Record Zero is written.

At the completion of command execution, the storage director is oriented to the Record Zero data area.

## Write Count, Key, and Data

Write Count, Key, and Data transfers data from the channel and formats a new record on the track. The first eight bytes transferred are the count area; the remaining bytes are written into the key and data areas.

Write CKD is valid in the domain of a Locate Record command that specified a Format Write (03) or Write Track (0B) Operation code. If it is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If a Write Count, Key, and Data command is immediately preceded by a Write Record Zero command to a defective track, it is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

When received outside the domain of a Locate Record, Write CKD must be preceded by either:

- A Search ID Equal or Search Key Equal command that compared equal on all bytes of the ID or key field, or
- A Write Record Zero command that executed outside the domain of a Locate Record, or
- A Write CKD command that executed outside the domain of a Locate Record.

If Write Count, Key, and Data is preceded by any other command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). A Read Key and Data, Read Data, Write Key and Data, or Write Data command may be executed between the Search ID Equal command and Write CKD; a Read Data or Write Data command may be executed between the Search Key Equal command and Write CKD. If the file mask inhibits format write operations (bits 0-1 = '01' or '10'), execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director requests an eight-byte count area from the channel. If the channel provides fewer than eight bytes, binary zeros are generated to fill the remaining bytes of the count area and the write operation continues as described below.

After the count area has been transferred, the storage director prepares to transfer the key and data areas from the channel. The key and data area lengths are defined by the count area.

If the key length (KL) is zero, the record is formatted without a key area.

If the data length (DL) is zero, the record is formatted as an end-of-file record.

The storage director attempts to transfer (KL + DL) bytes from the channel; if the channel provides fewer than (KL + DL) bytes, binary zeros are generated to fill the remaining bytes of the key and data areas.

Channel End and Device End status is presented when the write operation is complete at the device. If Write CKD is not followed by another Write CKD, the remainder of the track is erased after the record is written.

If the storage director determines that there is insufficient space on the track to write the complete record, execution is terminated with status that includes unit check (Invalid Track Format). No partial record will be left on the track.

**Programming Note:** The residual byte count in the SCSW will not necessarily indicate the amount by which the record length exceeded the available space on the track.

At the completion of command execution, the storage director is oriented to the data area of the record just written.

**Programming Note:** When the channel does not transfer a complete count area (for example, due to a CCW byte count less than 8) the format and Record ID of the record written on the track will be affected by the binary zeros generated by the control unit to fill the remaining bytes of the count area. If



the data length sub-field (bytes 6-7) of the count area contains zeros, the record written will be an end-of-file record.

## Write Count, Key, and Data Next Track

Write Count, Key, and Data Next Track causes data to be transferred from the channel and formatted as the first user data record on the *next* track. The first eight bytes transferred are the count area; the remaining bytes are written into the key and data areas.

Write Count, Key, and Data Next Track is valid only in the domain of a Locate Record command that specified a Format Write (03) Operation code.

The storage director will be oriented to a data area when execution of Write CKD Next Track is initiated; it erases the remainder of the current track, attempts to advance to the next track as described in "Operations within the domain of a Locate Record" on page 22, and prepares to format a user data record following Record Zero on the next track. If no Record Zero is detected on the next track, execution is terminated with status that includes unit check (No Record Found).

The storage director requests an eight-byte count area from the channel. If the channel provides fewer than eight bytes, binary zeros are generated to fill the remaining bytes of the count area and the write operation continues as described below.

After the count area has been transferred, the storage director prepares to transfer the key and data areas from the channel. The key and data area lengths are defined by the count area.

If the key length (KL) is zero, the record is formatted without a key area.

If the data length (DL) is zero, the record is formatted as an end-of-file record.

The storage director attempts to transfer (KL + DL) bytes from the channel; if the channel provides fewer than (KL + DL) bytes, binary zeros are generated to fill the remaining bytes of the key and data areas.

Channel End and Device End status is presented when the write operation is complete at the device. If Write CKD Next Track is not followed by a Write CKD, the remainder of the track is erased after the record is written.

If there is insufficient space on the track to write the complete record, execution is terminated with status that includes unit check (Invalid Track Format). No partial record will be left on the track. The residual byte count in the SCSW will not necessarily indicate the amount by which the record length exceeded the available space on the track.

At the completion of command execution, the storage director is oriented to the data area of the record just written.

**Programming Note:** When the channel does not transfer a complete count area (for example, due to a CCW byte count less than 8) the format and Record ID of the record written on the track will be affected by the binary zeros generated by the control unit to fill the remaining bytes of the count area. If the data length sub-field (bytes 6-7) of the count area contains zeros, the record written will be an end-of-file record.

## Erase

Erase overwrites any user data records for the remainder of the track.

Erase transfers data from the channel like Write CKD. However, instead of formatting a new record on the track, it discards the data transferred from the channel and erases the rest of the track, starting from the point at which Write CKD would have begun to write the record. The first eight bytes transferred are treated like a count area; the key length and data length sub-fields specify the number of additional bytes to be transferred.

Erase is valid in the domain of a Locate Record that specified a Write Track (0B) Operation code only. If it is received within the domain of a Locate Record that specified any other Operation code, it is rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

When received outside the domain of a Locate Record, Erase must be preceded by either:

- A Search ID Equal or Search Key Equal command that compared equal on all bytes of the ID or key field, or
- A Write Record Zero or Write CKD command that executed outside the domain of a Locate Record.

If Erase is preceded by any other command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). A Read Key and Data, Read Data, Write Key and Data, or Write Data command may be executed between the Search ID Equal command and Erase; a Read Data or Write Data command may be executed between the Search Key Equal command and Erase.

If the file mask inhibits format write operations (bits 0-1 = '01' or '10'), execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The rest of the track is erased starting from the end of the record written by the preceding Write R0 or Write CKD command, or from the end of the record that satisfied the Search ID Equal or Search Key Equal operation.

The storage director requests an eight-byte count area from the channel. If the channel provides fewer than eight bytes, binary zeros are generated to fill the remaining bytes and execution continues as described below.

After the count area has been transferred, the storage director prepares to transfer the key and data areas from the channel. The key and data area lengths are defined by the count area.

The storage director attempts to transfer (KL + DL) bytes from the channel; if the channel provides fewer than (KL + DL) bytes, data transfer is terminated.

Channel End status is presented when data transfer from the channel is completed. Device End status is presented when the erase operation is completed at the device.

If the storage director determines that there is insufficient space on the track to write a record of the size specified by the count area, execution is terminated with status that includes unit check (Invalid Track Format).

**Programming Note:** The residual byte count in the SCSW will not necessarily indicate the amount by which the record length exceeded the available space on the track.

## Write Update Key and Data

Write Update Key and Data transfers data from the channel and updates the key and data areas of an existing record.

Write Update Key and Data is only valid in the domain of a Locate Record command that specified a Write Data (01) Operation code. If it is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). If it is received outside the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the storage director is oriented to a count area when command execution is initiated (as the result of the execution of the preceding Locate Record command), it prepares to operate on the following key and data areas; otherwise (the preceding command was another Write Update Key and Data), the storage director orients to the next count area and prepares to operate on the key and data areas of that record. If End-of-Track is detected before a count area is found, execution continues as specified for the Locate Record Operation.

The storage director transfers data from the channel and writes it into the key and data areas of the record. The amount of data to be requested from the channel is defined by the Transfer Length Factor parameter of Locate Record (or by the Blocksize parameter of Define Extent if no TLF was specified). If the channel provides fewer than (KL + DL) bytes, binary zeros are generated to fill the remaining bytes of the key and data areas.

If the Transfer Length Factor (or Blocksize) value is not equal to the sum of the key length and data length fields in the count area of the record to be updated, the control unit action depends on whether Define Extent specified CKD Conversion mode in the Global Attribute parameter (byte 1, bit 2 = '1').

- When byte 1, bit 2 = '0' (not CKD Conversion mode), the record is not updated and the operation is terminated with status that includes unit check (Invalid Track Format).
- When byte 1, bit 2 = '1' (CKD Conversion mode), if the key length and data length are both zero, the record is not updated and execution is terminated with status that includes Unit Exception; otherwise, execution continues using the actual key and data lengths.

When the Transfer Length Factor (or Blocksize) value is equal to the sum of the key length and data length fields in the count area of the record to be updated, the key and data areas are updated. If the record being updated has a data area length of zero and CKD Conversion mode is specified, execution is then terminated with status that includes Unit Exception.

Channel End status is presented when data transfer from the channel is completed. If another write command is expected in the Locate Record domain, Device End status is included with Channel End; otherwise, Device End status is presented when the write operation is completed at the device.

At the completion of command execution, the storage director is oriented to the data area of the record that was just updated.

## Write Key and Data

Write Key and Data transfers data from the channel and updates the key and data areas of an existing record.

### Within the domain of a Locate Record

Write Key and Data is valid in the domain of a Locate Record command that specified a Write Data (01) Operation code and requires only one write operation (either a count of 1 or, with Read Count Suffix, a count of 2). If it is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director is oriented to a count area when command execution is initiated (as the result of the execution of the preceding Locate Record command) and it prepares to operate on the following key and data areas.

If End-of-Track is detected before a count area is found, execution continues as specified for the Locate Record operation.

The storage director transfers data from the channel and writes it into the key and data areas of the record. The amount of data to be requested from the channel is defined by the Transfer Length Factor parameter of Locate Record (or by the Blocksize parameter of Define Extent, if no TLF was specified). If the channel provides fewer than (KL + DL) bytes, binary zeros are generated to fill the remaining bytes of the key and data areas.

If the Transfer Length Factor (or Blocksize) value is not equal to the sum of the key length and data length fields in the count area of the record to be updated, the control unit action depends on whether Define Extent specified CKD Conversion mode in the Global Attribute parameter (byte 1, bit 2 = '1').

- When byte 1, bit 2 = '0' (not CKD Conversion mode), the record is not updated and the operation is terminated with status that includes unit check (Invalid Track Format).
- When byte 1, bit 2 = '1' (CKD Conversion mode), if the key length and data length are both zero, the record is not updated and execution is terminated with status that includes Unit Exception; otherwise, execution continues using the actual key and data lengths.

When the Transfer Length Factor (or Blocksize) value is equal to the sum of the key length and data length fields in the count area of the record to be updated, the key and data areas are updated. If the record being updated has a data area length of zero and CKD Conversion mode is specified, execution is then terminated with status that includes Unit Exception.

Channel End status is presented when data transfer from the channel is completed. Device End status is presented when the write operation is completed at the device.

If End-of-Track is detected within an area of, or between areas of the record to be updated, execution is terminated with status that includes unit check (Invalid Track Format).

At the completion of command execution, the storage director is oriented to the data area of the record that was just updated.

### **Outside the domain of a Locate Record**

The storage director is oriented to a count area when the command starts (as the result of the preceding Search ID command) and it writes the following key and data areas. The key and data area lengths are defined in the count area key length (KL) and data length (DL) fields. If the channel sends fewer than (KL) + (DL) bytes, binary zeros fill the remaining bytes of the key and data areas.

This command must be immediately preceded by a Search ID Equal command that compares equal on all bytes of the ID field. If it is not, the command is terminated with unit check status. The sense data contains command reject with format 0, message 2.

Unit check status with the sense data containing command reject with format 0, message 2 also occurs if the file mask does not permit update write operations (bits 0 and 1 are '01').

If the record to update has a data area length of zero, no data transfer occurs. The command is terminated with channel end, device end, and unit exception status.

If end-of-track is detected within the record, the operation is terminated with unit check status. The sense data includes Invalid Track Format.

The storage director presents channel end and device end status when the device completes the write operation. When the command completes, the storage director is oriented to the data area of the record just updated.

## **Write Update Data**

Write Update Data transfers data from the channel and updates the data area of an existing record.

Write Update Data is only valid in the domain of a Locate Record command that specified a Write Data (01) Operation code or in the domain of a Locate Record Extended that specified a Write Any (0A) Operation code. If Write Update Data is received within the domain of a Locate Record that specified any other Operation code, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). If Write Update Data is received outside the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the storage director is oriented to a count area when command execution is initiated (as the result of the execution of the preceding Locate Record command), it prepares to operate on the following data area; otherwise (the preceding

command was a Write Update Data), the storage director orients to the next count area and prepares to operate on the data area of that record. If End-of-Track is detected before a count area is found, execution proceeds as specified for the Locate Record operation.

The storage director transfers data from the channel and writes it into the data area of the record. The amount of data to be requested from the channel is defined by the Transfer Length Factor parameter of Locate Record (or by the Blocksize parameter of Define Extent, if no TLF was specified). If the channel provides fewer than DL bytes, binary zeros are generated to fill the remaining bytes of the data area.

If the record to be updated has a data area length of zero and CKD Conversion mode is specified (Define Extent Global Attributes byte, bit 2 = '1'), no data area transfer is attempted and execution is terminated with status that includes Unit Exception.

If the Transfer Length Factor (or Blocksize) value is not equal to the value in the data length field in the count area of the record to be updated, the control unit action depends on whether Define Extent specified CKD Conversion mode in the Global Attribute parameter (byte 1, bit 2 = '1').

- When byte 1, bit 2 = '0' (not CKD Conversion mode), the record is not updated and the operation is terminated with status that includes unit check (Invalid Track Format).
- When byte 1, bit 2 = '1' (CKD Conversion mode), if the data length is zero, the record is not updated and execution is terminated with status that includes Unit Exception; otherwise, execution continues using the actual key and data lengths.

Channel End status is presented when data transfer from the channel is completed. If another write command is expected in the Locate Record domain, Device End status is included with Channel End; otherwise, Device End status is presented when the write operation is completed at the device.

If End-of-Track is detected within an area of, or between areas of the record to be updated, execution is terminated with status that includes unit check (Invalid Track Format).

At the completion of command execution, the storage director is oriented to the data area of the record that was just updated.

## Write Data

Write Data transfers data from the channel and updates the data area of an existing record.

### Within the domain of a Locate Record

Write Data is valid in the domain of a Locate Record command that specified a Write Data (01) and requires only one write operation (either a count of 1 or, with Read Count Suffix, a count of 2) or a Write Track (0B) Operation code. If it is received within the domain of a Locate Record that specified any other operation, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Orientation is to a count area when the command is received (as the result of the preceding Locate Record command) and it prepares to operate on the following data area.

The storage director transfers data from the channel and writes it into the data area of the record. If the Locate Record operation is Write Track, the subsystem will assume a Transfer Length Factor of eight bytes for the R0 data field; otherwise, the amount of data to be requested from the channel is defined by the Transfer Length Factor parameter of Locate Record (or by the Blocksize parameter of Define Extent, if no TLF was specified). If the channel provides fewer than DL bytes, binary zeros are generated to fill the remaining bytes of the data area.

If the Transfer Length Factor (or Blocksize) value is not equal to the value in the data length field in the count area of the record to be updated, the control unit action depends on whether Define Extent specified CKD Conversion mode in the Global Attribute parameter (byte 1, bit 2 = '1').

- When byte 1, bit 2 = '0' (not CKD Conversion mode), the record is not updated and the operation is terminated with status that includes unit check (Invalid Track Format).
- When byte 1, bit 2 = '1' (CKD Conversion mode), if the data length is zero, the record is not updated and execution is terminated with status that includes Unit Exception; otherwise, execution continues using the actual key and data lengths.

Channel End status is presented when data transfer from the channel is completed. Device End status is presented when the write operation is completed at the device.

If End-of-Track is detected within an area of, or between areas of the record to be updated, execution is terminated with status that includes unit check (Invalid Track Format).

At the completion of command execution, the storage director is oriented to the data area of the record that was just updated.

### **Outside the domain of a Locate Record**

Write Data must be preceded by a Search ID Equal command that compared equal on all bytes of the ID field, or by a Search Key Equal command that compared equal on all bytes of the key area; otherwise, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the currently-accessed track is in the Device Support address space, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the file mask does not permit update write operations (bits 0-1 = '01'), execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The storage director is oriented to a count or key area when command execution is initiated (as the result of the execution of the preceding search command), and it prepares to operate on the following data area.

The data area length is defined by the data length (DL) field in the count area. The storage director attempts to transfer DL bytes from the channel; if the channel provides fewer than DL bytes, binary zeros are generated to fill the remaining bytes of the data area.

If the record to be updated has a data area length of zero, no data transfer occurs and execution is terminated with status that includes Channel End, Device End and Unit Exception.

Channel End and Device End status is presented when the write operation is complete at the device.

If End-of-Track is detected within an area of, or between areas of the record to be updated, execution is terminated with status that includes unit check (Invalid Track Format).

At the completion of command execution, the storage director is oriented to the data area of the record that was just updated.

---

## Sense commands

See Figure 5 on page 19 for a complete list of the 3990 commands.

## Sense

Sense causes 32 bytes of sense data to be transferred to the channel.

Sense is not valid in the domain of a Locate Record command. If a Sense command is received within the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Chapter 5, "Sense conditions" on page 169 contains a description of the sense data formats.

A contingent connection state is established for the device after the channel accepts a status byte containing Unit Check.

If the device is not busy, Sense will be executed, even if the device is in the not-ready state.

If the device is busy, Sense will be executed only if a contingent connection exists for the device on the channel path issuing the Sense command.

Channel End and Device End status is presented when data transfer is complete.

**Note:** System Reset or Selective Reset also terminate contingent connections to the channels on which they are received.

**Programming Note:** Reset Allegiance and Unconditional Reserve will clear outstanding contingent connections. Any channel that has a connection broken as a result of one of these commands will have Equipment Check, format X'3F', Reset Allegiance or Unconditional Reserve Issued sense pending in the storage facility. While Reset Allegiance will only execute if the device is not reserved to another path-group, Unconditional Reserve will always execute. In addition, Unconditional Reserve will return sense for



one outstanding contingent connection. If more than one contingent connection exists, the subsystem will select one of the sets of sense data and transfer it to the channel. The other sets of sense data will be discarded by the subsystem.

When a Sense command is issued to a device that is in the contingent connection state, much of the device status testing normally performed as part of initial selection is bypassed so that zero initial status can be presented and the error information relating to the last Unit Check can be reported. If any command immediately follows the Sense command in the same channel program and the device is still busy, execution of that command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 30).

**Programming Note:** The Sense command issued during a contingent connection to transmit the error description to the Host system should be a stand-alone command. The recommended ERP actions assume that it is (see Chapter 6, "Error recovery procedure" on page 203).

If other device commands are chained from the Sense command, the device may not be prepared to execute them. Since complete device status testing is performed for the commands that follow Sense, a non-zero initial status byte, that would normally be presented to the first CCW in a chain, may not be presented until some subsequent CCW, giving unpredictable results and requiring unique recovery actions.

If a program chooses to chain commands after the sense that breaks a contingent connection, the suggested ERP actions that are listed in Chapter 6, "Error recovery procedure" on page 203 may no longer be valid. The program must provide its own set of ERP actions.

## Sense ID

Sense ID causes 20 bytes of data describing the type and model number of the subsystem and device to be transferred to the channel.

Sense ID is not valid in the domain of a Locate Record command. If a Sense ID command is received within the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the device is not busy, Sense ID will be executed even if the device is in the not ready state or absent.

### Byte Definition

<b>0</b>	X'FF'
<b>1-2</b>	Storage Director type
<b>3</b>	Storage Director model
<b>4-5</b>	Device type
<b>6</b>	Device model
<b>7</b>	Zero
<b>8-11</b>	CIW for Read Configuration Data
<b>12-15</b>	CIW for Set-Interface-Identifier
<b>16-19</b>	CIW for Read-Node-Identifier

Bytes 1-6 are the same as bytes 0-5 that the Read Device Characteristics command transfers.

Bytes 7-19 contain extended-identification information. Byte 7 is reserved. Bytes 8-19 contain command information words (four bytes each) that describe a command that will cause the storage facility to perform a specific operation. Each command information word contains the following:

- A *command type* that will perform a specific operation. The command types that can be used for the ESS subsystem are:
  - Read Configuration Data (RCD): this command type obtains configuration-dependent information.
  - Set Interface Identifier (SII): this command type identifies the interface used to determine a node ID.
  - Read node identifier (RNI): this command type is used to obtain node identification information
- A *command code* used to cause the volume to perform the operation specified by the command type. The command codes specified for the ESS storage facility are:
  - X'FA' – Read Configuration Data (RCD)
  - X'27' – Set Interface Identifier (SII)
  - X'3E' – Read Node Identifier (RNI)
- A *count* that specifies the number of bytes transferred

See *ESA/390 Common I/O Device Commands, SA22-7204* for more information.

See Figure 62 on page 221 for actual values.

Channel End and Device End status are presented when data transfer is complete.

---

## Miscellaneous commands

The commands described in this section are not a part of ECKD, but are implemented by this storage subsystem.

See Figure 5 on page 19 for a complete list of channel commands.

## No-Operation

No-Operation checks the logical device status and causes no action to be performed at the device.

No-Op is not valid in the domain of a Locate Record command. If a No-Op command is received within the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

No-Op is processed as an immediate command with Channel End and Device End presented in initial status.

No-Op resets orientation and should not be used indiscriminately. No-Op inserted within CCW chains may cause records or parts of records to be skipped. This loss

of orientation will occur when operating with cache as well as DASD. For example, a No-Op inserted between a Read Count and a Read Data causes the data area of some subsequent record to be read. Also, a No-Op inserted between a command that reads the data area of record (N-1) and a command that must process the count area of record (N) may skip one or more records and process the count area of a subsequent record. In this situation, the channel turnaround time, the record lengths, and other internal subsystem variables dictate which record is processed.

Outstanding sense information for the channel path is not disturbed unless Unit Check occurs. If Unit Check occurs, the new sense information replaces any previous sense information being held for the device on the selected channel path.

**Programming Note:** This command does not transfer any data. Therefore, if the CCW specifies a non-zero byte count, the SLI bit should be set to '1' to avoid an incorrect length exception.

## Read and Reset Buffered Log

Read and Reset Buffered Log causes 32 bytes of usage/error information to be transferred from the subsystem to the channel.

Read and Reset Buffered Log is not valid in the domain of a Locate Record command; if Read and Reset Buffered Log is received within the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The format of the 32 bytes is shown in the following table.

*Figure 17. Read and Reset Buffered Log*

Bytes	Contents
0-1	Zero.
2	Set to the Control Unit Type Code that appears in byte 42 of the data returned by the Read Device Characteristics command.
3	Zero
4	Logical Device Unit Address
5	Set to the Device Type Code that appears in byte 57 of the data returned by the Read Device Characteristics command.
6	Content and Format - set to X'41', as defined in "Sense Byte 6, Content and Format" on page 192.
7-9	zero.
10-14	storage facility Serial Number (compressed, as defined in "Sense Bytes 15-19, Serial Number" on page 192).
15-19	storage facility Serial Number (compressed, as defined in "Sense Bytes 15-19, Serial Number" on page 192).
20-21	Subsystem ID (SSID) of the affected LSS.
22	Exception Class, set to X'6F', as defined in "Sense Bytes 22-23, Exception Code" on page 193.
23	Exception Code, set to zeros.
24	Flags - set to zeros, as defined in "Sense Byte 24, Flags" on page 193.
25	Program Action Code, set to zeros, as defined in "Sense Byte 25, Program Action Code" on page 194.
26-27	Configuration Data, as defined in "Sense Byte 26-27, Configuration Data" on page 194.
28-31	Zeros.

The statistics may or may not reflect the activity accumulated during the chain that includes the Read and Reset Buffered Log command.

Channel End is presented when the data transfer is complete. Device End is presented when the counters have been reset. Channel End and Device End may be presented together.

## Diagnostic Control

Diagnostic Control transfers a variable number of parameter bytes from the channel. The parameters specify a service or information request.

If Diagnostic Control is received in the domain of a Locate Record command, it is rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If a Perform Subsystem Function, Set Special Intercept Condition was executed for the addressed device on any interface with the same path group identifier as the interface processing the command, processing of this command is terminated with status that includes unit check (Command Reject, Environmental Data Present, format X'0F' Reason Code X'80') with a program action code of X'70'. See "Set Special Intercept Condition (Order X'1B') on page 118.

The control unit attempts to transfer four bytes of parameters from the channel; after verifying these bytes, the control unit will attempt to transfer any additional bytes required (as specified in bytes 2-3); if the channel transfers fewer bytes than expected, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

The first four bytes transferred from the channel have the following format:

- 0**            Subcommand
- 1**            Subcommand modifier
- 2-3**         Number of additional parameter bytes to be transferred

Published subcommands and their requirements are defined in Figure 18. Other subcommands may be defined for a particular control unit model and may not be published.

If a subcommand is not valid for the device or the Control Unit model it was received on, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If the value in bytes 2-3 is not what is expected for the subcommand, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

Some subcommands of this command are accepted on a subsystem that has not located Global subsystem status devices and on devices with an unreadable Partition Status Area (X'FC' state). If the subsystem or device is in X'FC' state and a subcommand that is not accepted when in this state is received, the command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 1D).

Other parameter checks and chaining requirements are dependent on the subcommand and are defined in the subcommand sections that follow.

Some subcommands cannot be preceded in the CCW chain by a command that specifies cylinder and head selection (Locate Record, Locate Record Extended, Seek, Seek Cylinder, or Read IPL). If this condition is not met, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Some subcommands place restrictions on the commands that follow them. If a command that is not allowed (as defined in Figure 18) follows this command, it is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If Diagnostic Access authorization or Device Support authorization is required and is not met, the command is terminated with status that includes unit check (Command Reject, format X'05', Diagnostic Command Violates File Mask).

If the subcommands must be the first command in the CCW chain or chained directly from a Suspend Multipath Reconnection that is the first command in the chain and that condition is not met, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Figure 18. Diagnostic Control subcommands and their chaining and parameter requirements

Byte 0	Subcommand	Diagnostic or Device Support Authorization required?	Must be First in Chain or Chained from SMR that is FIC?	Accepted in X'FC' state?	Devices accepted on:	Can be preceded by a command that specifies cylinder and head selection?	Commands that can follow this command:
02	Inhibit Write	Yes	No	No	All	No	Any <sup>1</sup>
08	Enable Write	Yes	No	No	All	No	Any <sup>1</sup>
09	Emulation Control	Yes	No	No	All	No	Any <sup>1</sup>
0B	Diagnostic Initialize Subsystem	No	Yes	Yes	All	No	None
0C	Unfence	Yes	No	No	All	No	Any <sup>1</sup>
14	Force State Save	No	Yes <sup>2</sup>	Yes	All	No	None

**Notes:**

<sup>1</sup> Any command that does not have further restrictions such as being first in chain (like SPID).

<sup>2</sup> Force state save can be issued standalone and will be accepted even if a subsystem or device long busy condition exists.

**All listed Diagnostic Control subcommands:**

- Have bytes 2-3 = 0.
- Are expected to be four bytes in length.
- May NOT be preceded by a Remote Service Access order.

Channel End status is presented after a valid parameter list has been received. Device End status is presented when the operation has been completed unless RAS Microcode access is required. Exceptions to Device End presentation will be noted in subcommands requiring RAS Microcode access. Channel End and Device End may be presented together.

**Inhibit Write - 02**

If the subcommand is 'Inhibit Write', the Storage Director will set controls to reject any further write operations through the path defined in parameter byte 1, the subcommand modifier. This byte is defined as follows:

**Byte 1 Description**

**40** Inhibit Write operations to all devices from the selected logical path.

The subcommand modifier must be one of the specified values or the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

After this command has been accepted all write commands and all Locate Record commands specifying write type operations received on an inhibited path will be rejected with unit check. Equipment Check (byte 0, bit 3) and Write Inhibited (byte 1, bit 6) will be included in sense data. Once write operations have been inhibited on a path, the path will remain inhibited until a Diagnostic Control command with the 'Enable Write' (08) subcommand is received or it is reset by system reset or IML or Restart Storage Control. System reset will reset inhibit write for the channel path within the Storage Director receiving the reset. IML and restart Storage Control reset all Inhibit write controls within the Storage Director.

### **Enable Write - 08**

If the subcommand is 'Enable Write', the Storage Director will reset the controls set by any previous 'Inhibit Write' subcommands received by this Storage Director.

Parameter byte 1 must be zeros or the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

### **Track Emulation Control - 09**

This subcommand controls switching the addressed device between native and emulation mode.

The operation is dependent on the subcommand modifier in byte 1.

#### **Value Description**

**00** Display Native Device

**01-FF** Reserved

Parameter byte 1 must be one of the specified values or the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

The operation is subcommand dependent:

**Display Native Device:** This subcommand will be accepted only if the device is Track Emulation Mode or in transition to Track Emulation Mode. If this condition is not met, the operation is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 3B).

If the subcommand modifier is '00' and the immediately following command is a Read Device Characteristics or Sense ID command, the data returned will be for the Native device. This subcommand has no effect on any other commands.

Channel End and Device End are presented when the operation is complete.

### **Diagnostic Initialize Subsystem - 0B**

This subcommand will cause the subsystem to establish or reestablish all caching status on the Subsystem Status Partition. It is intended as a reconfiguration and recovery tool.

The specific action is determined by the subcommand modifier in byte 1 as follows:

#### **Value Description**

**00** Conditional Reset and Establish Global subsystem status

**01** Unconditional Reset and Establish Global subsystem status.

**02-FF** Reserved

Byte 1 must contain one of the valid values or the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If a previous Diagnostic Initialize Subsystem has failed to complete when this order is received, this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 2C).

The activity is dependent on the subcommand modifier.

**Conditional Reset and Establish, '00':** If the subsystem has found a valid Global subsystem status track, the command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 20).

This subcommand with a '00' modifier will cause a ESS subsystem to reset all caching status to its default values and to establish a new set of Subsystem Status Partition. It will cause all data in non-volatile storage to be lost. The CFWID and Global subsystem status reconfiguration count are incremented by X'10' from the highest count found.

The primary intent of this command is to allow a customer to bring a subsystem up after a configuration change, or if a catastrophic error prevents the subsystem from finding a valid Global subsystem status track.

This command will require that both IML devices be updated to point to the Subsystem Status Partition.

Channel End and Device End status is presented when the parameters are received and the command is accepted.

When the operation is complete a state change interrupt is set pending for all devices in the subsystem and the appropriate termination and reinitialization sense messages will be presented as Environmental Data.

**Unconditional Reset and Establish, 01:** This function is identical to the '00' subcommand modifier except that it will reset and establish the Global subsystem status, even if valid Global subsystem status currently exist in the subsystem.

### **Unfence - 0C**

This subcommand causes fences to be removed.

Parameter byte 1 bits 5-7 must be zero and at least one bit must be set in bits 0-4. If bit 4 is set, no other bits may be set. If these conditions are not met, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

Channel End and Device End are presented after parameters have been verified. The subsystem then takes action to reset all path related fences within the Storage Controller.

Channel operations queued in the Storage Control may be terminated with status that includes unit check (Equipment Check, format 0, message 9) as a result of the 'Restart Storage Control'.

### **Force State Save - 14**

The Force State Save order causes the controller to perform a state save. If there is a state save in progress when this order is received, the command will be accepted but will not cause another state save operation.

This order transfers 20 bytes of data, defined as follows:

#### **Byte Value**

- |          |    |                                 |
|----------|----|---------------------------------|
| <b>0</b> | 14 | Force State Save                |
| <b>1</b> | 00 | Indicates a generic state save. |



4-19 Unused.

## Read Configuration Data

The Read Configuration Data command causes 256 bytes of configuration data to be transferred to the channel.

Read Configuration Data is not valid in the domain of a Locate Record command; if Read Configuration Data is received within the domain of a Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Figure 19. Configuration data

Byte	Description
0-31	NED (Node Element Descriptor) 1; contains the NED for the device
32-63	NED 2; contains the NED for the RAID Rank
64-95	NED 3; contains the NED for the Cluster that the SA resides in
96-127	NED 4; contains the token NED for the subsystem
128-223	Reserved
224-255	Node Element Qualifier (NEQ)

Channel End and Device End status is presented when data transfer is complete.

### NED 1, the NED for the volume

Byte	Description
0	Flags <b>Bit Description</b> <b>0-1</b> Field Identifier <b>00</b> Field unused <b>01</b> Specific NEQ <b>10</b> General NEQ <b>11</b> NED These bits are set to 11. <b>2</b> Token Indicator, set to zero. <b>3</b> Serial number valid, set to one <b>4</b> Substitute Serial number, set to one <b>5</b> Reconfiguration NED, Set to one. <b>6</b> Emulation NED, Set to Zero. <b>7</b> Must be zero.
1	NED Type, = 1 => I/O device
2	Class, = 1 => Direct Access Storage Device
3	Bits 0-6: Must be zero. Bit 7: Level, set to zero.
4-9	Device Type, in EBCDIC = '2105'
10-12	Device Model, in EBCDIC = ' '
13-15	HDA manufacturer, = "IBM"
16-17	HDA manufacturing location, in EBCDIC
18-29	storage facility sequence number, in EBCDIC. The valid digits for the box are right justified in the field with leading EBCDIC zeroes (F0) padding.
30-31	Tag

## NED 2, the NED for the RAID rank

Byte	Description
0	Flags <b>Bit Description</b> <b>0-1</b> Field Identifier <b>00</b> Reserved <b>01</b> Specific NEQ <b>10</b> General NEQ <b>11</b> NED These bits are set to 11 (NED). <b>2</b> Token Indicator, set to zero for ESS. <b>3</b> Serial number Valid, set to one <b>4</b> Substitute Serial number, set to one <b>5</b> Reconfiguration NED, Set to one. <b>6</b> Emulation NED, Set to Zero. <b>7</b> Must be zero.
1	NED Type, = 0 => Unspecified type
2	Must be zero.
3	Bits 0-6: Must be zero. Bit 7: Level, set to zero.
4-9	Rank Type, in EBCDIC = '2105'
10-12	Rank Model, in EBCDIC = ' ' ' ' ' '
13-15	Drawer manufacturer, = "IBM"
16-17	Drawer manufacturing location, in EBCDIC
18-29	storage facility sequence number, in EBCDIC. The valid digits for the box are right justified in the field with leading EBCDIC zeroes (F0) padding.
30-31	Tag

## NED 3, the NED for the SA I/O board

Byte	Description
0	Flags <b>Bit Description</b> <b>0-1</b> Field Identifier <b>00</b> Reserved <b>01</b> Specific NEQ <b>10</b> General NEQ <b>11</b> NED These bits are set to 11 (NED). <b>2</b> Token Indicator, set to zero. <b>3</b> Serial number valid, set to one. <b>4</b> Substitute Serial number, set to zero. <b>5</b> Reconfiguration NED, Set to one. <b>6</b> Emulation NED, Set to Zero. <b>7</b> Must be zero.
1	NED Type, = 2 => Control Unit Type
2	Must be zero.
3	Bits 0-6: Must be zero. Bit 7: Level, set to zero.
4-9	storage facility Type, in EBCDIC = "2105"
10-12	storage facility Model, in EBCDIC
13-15	storage facility manufacturer, = "IBM"
16-17	storage facility manufacturing location, in EBCDIC
18-29	storage facility sequence number, in EBCDIC
30-31	Tag

## NED 4, the token NED for the subsystem

Byte	Description																										
0	Flags  <table border="0"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0-1</td> <td>Field Identifier</td> </tr> <tr> <td>00</td> <td>Reserved</td> </tr> <tr> <td>01</td> <td>Specific NEQ</td> </tr> <tr> <td>10</td> <td>General NEQ</td> </tr> <tr> <td>11</td> <td>NED</td> </tr> <tr> <td></td> <td>These bits are set to 11.</td> </tr> <tr> <td>2</td> <td>Token Indicator, set to one.</td> </tr> <tr> <td>3</td> <td>Serial number valid, set to one.</td> </tr> <tr> <td>4</td> <td>Substitute Serial number, set to zero.</td> </tr> <tr> <td>5</td> <td>Reconfiguration NED, Set to zero.</td> </tr> <tr> <td>6</td> <td>Emulation NED, Set to Zero.</td> </tr> <tr> <td>7</td> <td>Must be zero.</td> </tr> </tbody> </table>	Bit	Description	0-1	Field Identifier	00	Reserved	01	Specific NEQ	10	General NEQ	11	NED		These bits are set to 11.	2	Token Indicator, set to one.	3	Serial number valid, set to one.	4	Substitute Serial number, set to zero.	5	Reconfiguration NED, Set to zero.	6	Emulation NED, Set to Zero.	7	Must be zero.
Bit	Description																										
0-1	Field Identifier																										
00	Reserved																										
01	Specific NEQ																										
10	General NEQ																										
11	NED																										
	These bits are set to 11.																										
2	Token Indicator, set to one.																										
3	Serial number valid, set to one.																										
4	Substitute Serial number, set to zero.																										
5	Reconfiguration NED, Set to zero.																										
6	Emulation NED, Set to Zero.																										
7	Must be zero.																										
1	NED Type, = 0 => unspecified																										
2	Must be zero.																										
3	Bits 0-6 Must be zero.  Bit 7: Level, set to one.																										
4-9	storage facility Type, in EBCDIC = "2105"																										
10-12	Model Number.  The field is set to all blanks in EBCDIC.																										
13-15	storage facility manufacturer, = "IBM"																										
16-17	storage facility manufacturing location, in EBCDIC																										
18-29	storage facility sequence number in EBCDIC																										
30-31	Tag																										

## Node Element Qualifier (NEQ)

Byte	Description																
0	Flags  <table border="0"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0-1</td> <td>Field Identifier</td> </tr> <tr> <td>00</td> <td>Reserved</td> </tr> <tr> <td>01</td> <td>Specific NEQ</td> </tr> <tr> <td>10</td> <td>General NEQ</td> </tr> <tr> <td>11</td> <td>NED</td> </tr> <tr> <td></td> <td>These bits are set to 10 (General NEQ).</td> </tr> <tr> <td>2-7</td> <td>Must be zero.</td> </tr> </tbody> </table>	Bit	Description	0-1	Field Identifier	00	Reserved	01	Specific NEQ	10	General NEQ	11	NED		These bits are set to 10 (General NEQ).	2-7	Must be zero.
Bit	Description																
0-1	Field Identifier																
00	Reserved																
01	Specific NEQ																
10	General NEQ																
11	NED																
	These bits are set to 10 (General NEQ).																
2-7	Must be zero.																
1	Record Selector, always zero for ESS.																
2-3	Interface ID  The first byte (2) is zero. The second byte is defined as follows:  <table border="0"> <thead> <tr> <th>Bit</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Storage Cluster number</td> </tr> <tr> <td>1</td> <td>Zero.</td> </tr> <tr> <td>2</td> <td>I/O board number</td> </tr> <tr> <td>3</td> <td>Zero.</td> </tr> <tr> <td>4-5</td> <td>Card slot number</td> </tr> <tr> <td>6</td> <td>Zero</td> </tr> <tr> <td>7</td> <td>Port number on the card.</td> </tr> </tbody> </table>	Bit	Definition	0	Storage Cluster number	1	Zero.	2	I/O board number	3	Zero.	4-5	Card slot number	6	Zero	7	Port number on the card.
Bit	Definition																
0	Storage Cluster number																
1	Zero.																
2	I/O board number																
3	Zero.																
4-5	Card slot number																
6	Zero																
7	Port number on the card.																
4-5	Must be zero																

Byte	Description
6	<p>Primary Missing Interrupt Timer Interval</p> <p>This byte is used to specify a default missing interrupt timer interval.</p> <p><b>Bit Description</b></p> <p><b>0-1</b> Increment Value These bits define the size of the increment to count.</p> <p><b>Value Definition</b></p> <p><b>00</b> Count in 1 second increments  <b>01</b> Count in 10 second increments  <b>10</b> Count in 100 second increments  <b>11</b> Count in 1000 second increments</p> <p><b>2-7</b> Number of increments A value of zero indicates no default interval has been specified.</p> <p>ESS will set this byte to X'1E' (30). This will specify that the default Missing Interrupt Timer Interval should be 30 one-second increments, or 30 seconds.</p>
7	<p>Secondary Missing Interrupt Timer Interval</p> <p>This is used to specify a secondary interval. It is used by some I/O devices that have a different interval for certain commands. The definition of the value of the interval is the same as byte 6 above.</p> <p>ESS does not use this interval and will set the byte to zero.</p>
8-9	Reserved
10	Number of LSA RAID ranks configured to subsystem. For Home area subsystems this is a zero.
11	Unit Address, the address as seen by the Storage Director on the channel interface.
12	Physical Device ID. For ESS this is set to be the same as byte 11
13	Physical Device Address. For ESS this is set to be the same as byte 11
14	SA ID (same as interface ID, byte 3)
15-16	<p>ESCON/serial Link Address</p> <p><b>Byte 15</b> Host Channel link address  <b>Byte 16</b> Host Channel logical address</p> <p>These values are derived from the ESCON command frame received from the host that issued the Read Configuration Data command.</p>
17	<p>Interface Protocol Type</p> <p><b>Bit Definition when bit=1</b></p> <p><b>0</b> Parallel channel (including 9034), set to zero by ESS  <b>1</b> ESCON channel, set to one  <b>2-7</b> Reserved</p>
18	<p>NEQ format flags</p> <p><b>Bit Definition when bit=1</b></p> <p><b>0</b> Byte 12 is Physical Device ID, byte 19 is LDA. Set to ONE by ESS.  <b>1</b> Number of Logical Paths supported is in bytes 30-31.  <b>2-7</b> Reserved</p>
19	Logical Device Address. For ESS this is set to be the same as byte 11
20-23	Reserved
24-29	Must be zero
30-31	Number of Logical Paths supported by this storage facility Set to 64 times the number of configured ESCON ports, plus 256 times the number of configured Fibre channel ports

---

## Path control commands

See Figure 5 on page 19 for a complete list of channel commands.

### Device Reserve

Device Reserve causes 32 bytes of sense data to be transferred to the channel.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Device Reserve must be the first command in the channel program or chained directly from a Suspend Multipath Reconnection that was the first command in the chain. If any other command precedes Device Reserve, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The Device Reserve command will not be executed if the logical device is reserved to another path group.

If the logical device is not busy, Device Reserve will be executed even if the device is in the not-ready state.

The Device Reserve command includes all the functions of the Sense command (see "Sense" on page 86), and in addition, causes the addressed device to become reserved to the channel path group or path of the channel issuing the command. If the command is received on a device that is grouped in either single-path or multi-path mode, it is reserved to the path group. If the command is received over a channel path for which the device is not grouped, the addressed device will be reserved to that channel path. The device reservation lasts until that channel or any channel in the channel path group successfully completes a Device Release command against the device or all paths on which the device is reserved are reset by a system reset or a subsystem IMPL.

When a Device Reserve command is issued to a device that is in the contingent connection state, much of the device status testing normally performed as part of initial selection is bypassed so that zero initial status can be presented and the error information relating to the last Unit Check can be reported.

The following list outlines actions that are taken when the subsystem receives a Device Reserve command.

- If there are no channel programs outstanding for the device, the device state is set to reserved allegiance. The subsystem presents Channel End and Device End after it transfers 32 sense bytes.
- If there are channel programs outstanding for the device, delayed command retry status is presented and the device state is set to pending reserve. The device is not considered reserved while in that state. After all of the outstanding channel programs for the device have completed the disconnected channel program is reconnected and the Device Reserve command is issued again. At this time the device state is set to reserved allegiance and Channel End and Device End are presented following the transfer of 32 sense bytes.

If the channel ends command retry, the pending reserve state is reset at the device, and the Device Reserve command is terminated with status that includes unit check, Attachment Check 02.

- If the device state is either reserved allegiance or pending reserve for another path or path group, the subsystem returns device busy status in response to the Device Reserve command.
- New I/O requests from paths or path groups other than those holding the reserved allegiance or pending reserve will receive device busy status. Commands that bypass device busy status will be processed.

**Note:** Device reservation is reset for an individual channel path if a storage director receives a System Reset from that channel.

**Note:** If a device is reserved to a channel path or channel path group, the existing device reservation is reset for that channel path or channel path group when Unconditional Reserve is successfully completed by any channel attached to the storage subsystem. The device becomes reserved to the channel path or path group that issued the Unconditional Reserve Command.

## Device Release

Device Release causes 32 bytes of sense data to be transferred to the channel.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If Device Release was preceded in the command chain by a Define Extent, a Set File Mask, or a Read IPL, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The Device Release command will not be executed if the device is busy.

If the device is not busy, Device Release will be executed even if the device is in the not-ready or 'Status Cannot be Determined' state.

The Device Release command includes all the functions of the Sense command (see "Sense" on page 86). In addition, if the command is received over a channel path with the device in the grouped state, it causes reservation of the addressed device to be terminated for the channel path group of the channel issuing the command. If the command is received over a channel path that is in an ungrouped state, it will cause the reservation of the addressed device to be terminated for that channel path.

When a Device Release command is issued to a device that is in the contingent connection state, much of the device status testing normally performed as part of initial selection is bypassed so that zero initial status can be presented and the error information relating to the last Unit Check can be reported.

## Unconditional Reserve

Unconditional Reserve causes 32 bytes of sense data to be transferred to the channel.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Unconditional Reserve must be the first command in the channel program or chained directly from a Suspend Multipath Reconnection that was the first command in the chain. If any other command precedes Unconditional Reserve, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Unconditional Reserve will be executed even if the device is in either the busy or not-ready state.

The purpose of the Unconditional Reserve command is to break a device's allegiance or reservation to a channel path or path group. The intent is to establish reservation to an alternate channel path within the same Host system.

The Unconditional Reserve command includes all the functions of the Sense command (see "Sense" on page 86) and in addition, causes the addressed device to become reserved to the channel path group or path of the channel issuing the command. If the command is received on a device that is grouped in either single-path or multi-path mode, it is reserved to the path group. If the command is received over a channel path for which the device is not grouped, the addressed device will be reserved to that channel path. The device reservation lasts until that channel or any channel in the channel path group successfully completes a Device Release command against the device or all paths on which the device is reserved are reset by a system reset or a subsystem IMPL.

If the host program does not want to reserve the device, it must chain a Device Release command to the Unconditional Reserve command.

**Attention:** The System Control Program should use great care when using the Unconditional Reserve command. Issuing this command can interrupt other I/O operations on this system image and other system images.

The Unconditional Reserve command may cause the following effects on *other* channel paths or path groups in the *same* system image or in *other* host system images.

- If the device is reserved, the reservation is reset and the device becomes reserved as described above.
- Started operations are terminated with status that includes unit check (Equipment Check, format X'3F', Reset Allegiance or Unconditional Reserve Issued).
- If the channel path or path group is idle and the volume is not reserved, there is no effect.

The following list outlines actions that are taken when the control unit receives an Unconditional Reserve command:

- If there is a channel program outstanding for the device on the channel path or path group issuing the Unconditional Reserve command, the control unit will present Busy status to the UR command, perform an internal selective reset, then present Device End status to get the UR chain redriven.
- If there is no channel program outstanding nor a contingent allegiance for the device on the channel path or path group issuing the Unconditional Reserve command, null sense data (format x'00', No Message) is returned to the channel.
- If there is a contingent allegiance for the same channel path or path group that issued the Unconditional Reserve, the associated sense data is returned and the contingent allegiance is reset.
- If there are channel programs outstanding for the device on channel paths or path groups other than the one issuing the Unconditional Reserve command, each of those operations is terminated with status that includes unit check (Equipment Check, format X'3F', Reset Allegiance or Unconditional Reserve Issued).
- If the device has a reserve allegiance for another channel path or path group that has no I/O operations outstanding, then the next channel initiated operation from that channel path or path group is terminated with status that includes unit check (Equipment Check, format X'3F', Reset Allegiance or Unconditional Reserve Issued).
- If there was a contingent allegiance for another channel path or path group prior to the reception of the Unconditional Reserve command and the associated Sense command is received, the Sense command is terminated with status that includes unit check (Equipment Check, format X'3F', Reset Allegiance or Unconditional Reserve Issued).

**Note:** In this case the original sense data is replaced with the Equipment Check, format X'3F', Reset Allegiance or Unconditional Reserve Issued sense data.

The control unit presents Channel End and Device End after it transfers the 32 bytes.

## Reset Allegiance

The purpose of the Reset Allegiance command is to break a device's allegiance to one or more channel paths or path groups. It will be executed even if the device is in either the busy or not-ready state. However it will not reset any allegiances if the device is reserved to another channel or path group.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Reset Allegiance must be the first command in the channel program or chained directly from a Suspend Multipath Reconnection that was the first CCW in the chain. If any other command precedes Reset Allegiance, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Reset Allegiance causes 32 bytes of data to be transferred to the channel. This data is defined as follows:



Byte	Bit	Definition	
0	Status		
	0	Device selection successful. When this bit is zero (device selection failed) and bits 2-3 are not 01, the Storage Director attempted device selection and failed. The failure that prevented the selection will be reported as Environmental Data to the next initial selection to the failed device.	
	1	Must be zero	
	2-3	Reservation status	
		00	Device is not reserved.
		01	Device is reserved to another channel or path-group.
		10	Device is reserved to this channel or path-group.
	4-5	Allegiance reset for the addressed device.	
00		No allegiance was reset for the addressed device.	
01		Implicit allegiance was reset for the addressed device.	
	10	Contingent allegiance was reset for the addressed device.	
	6	Contingent allegiance was reset on another device.	
	7	Implicit allegiance was reset on another device.	
1	0-1	Partitioning Status, always zero for ESS	
	2	Path Group Identifier Supported, set to one	
	3-7	Must be zero	
2-4	Must be zero		
5-15	Path Group Identifier If byte 0 bits 2-3 = 01 => device reserved to another channel or path group and a PGID exists (the channel / path group that the device is reserved to has valid PGID), these bytes will contain the Path Group ID. Otherwise these bytes will be zero.		
16- 31	Must be zero		

If the device is either not reserved or reserved to this channel or path-group, the following occurs:

- Any implicit allegiances for this device are reset.
- Any CCW chain in progress on this device for this channel or path-group is terminated and any status owed for that CCW chain including sense is reset. However, multi-tagged or untagged status for this device is not reset.
- Any CCW chain in progress on this device for a different channel or path-group will be terminated with status that includes unit check (Equipment Check, format X'3F', Reset Allegiance or Unconditional Reserve Issued).

**Note:** If the device is currently selected on another channel, it may be terminated with either status that includes unit check or a Disconnect sequence.

- Any sense pending on this device for a different channel or path-group will be changed to Equipment Check, format X'3F', Reset Allegiance or Unconditional Reserve Issued.
- The device will be selected, if possible, using unconditional selection sequences, if necessary.
- 32 bytes of data are passed back to the host.
- If selection was successful, any CCW that can be chained to another CCW may be chained to this one.

If the device is reserved to another channel or path-group, Reset Allegiance will return the 32 bytes of data and do nothing more. In this case any CCW chained from this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 2F).

## Set Path Group ID (SPID)

A 12-byte parameter is transferred from the channel.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Set Path Group ID must be the only command in the channel program or chained directly from a Suspend Multipath Reconnection that was the first command in the channel program. If Set Path Group ID is preceded by any other command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence); if any command follows Set Path Group ID in a channel program, that command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

This command will be executed even if the device is busy or in the not-ready or 'Status Cannot be Determined' state.

Set Path Group ID causes 12 bytes to be transferred to the storage subsystem. The first byte is a function control byte and the following 11 bytes are the Path Group ID. The Path Group ID must be assured unique by the using Host systems to assure integrity of the data. For example, if two system control programs use the same Path Group ID, a reserved device will be available to both. In addition, if two system control programs use the same Path Group ID in a multi-path environment, the results will be status handling malfunctions.

The subsystem requests 12 bytes of parameters from the channel; if the channel provides fewer than 12 bytes, execution is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

Byte 0 is the function control byte; bytes 1 through 11 contain the Path Group ID.

At the completion of parameter transfer, the subsystem inspects the data for validity. The command may be rejected for the following reasons:

1. Path Group ID contains 11 bytes of zeros.
2. A path Group ID is already established for a channel path and the Path Group ID parameters in this command are not the same.
3. A Mode bit is already established for a channel path and the Mode bit specified in this command is not the same.
4. The group code is all ones (function control byte bits 1,2 = 11).
5. Bits 3-7 of the function control byte are not zeros.

If an invalid parameter is received from the channel, execution is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If this command would cause more than 64 paths in the path group, this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 5D).

If this command would cause more than 128 path groups to be created in the subsystem, this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 5E).

Figure 20. Control byte definition

Bits	Value	Definition
0	0	Single Path Mode I/O operations must be maintained for the specific channel path over which the start I/O was initiated. Grouping exists only for reservations, not for reconnection.
	1	Multi-path Mode I/O operations started on one channel path may reconnect and continue or complete on any path in the group. Grouping applies to device reservations and reconnection.
1-2	00	Establish Group A group is formed for the addressed device consisting of all channel paths with matching IDs.
	01	Disband Group Each channel path for this device is placed in the ungrouped state. If the addressed device was reserved to the channel path group when the command was received, the device remains reserved only to the channel that issued the command.
	10	Resign from Group The issuing channel is removed from the channel path group for the device receiving the command. If the addressed device was reserved to the channel path group when the command was received, the reservation remains with the remaining members of the group.
	11	Reserved
3-7		Must be zero

## Sense Path Group ID (SNID)

Sense Path Group ID causes 12 bytes of path group data to be transferred to the channel.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Sense Path Group ID must be the only command in the channel program or chained directly from a Suspend Multipath Reconnection that was the first command in the channel program. If Sense Path Group ID is preceded by any other command, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence); if any command follows Sense Path Group ID in a channel program, that command will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

This command will be executed, even if the device is busy or in the not-ready or 'Status Cannot be Determined' state.

**Note:** This command can execute while the device is busy. (see “Sense” on page 86 for ramifications of bypassing the device busy indication.)

Sense Path Group ID causes 12 bytes of path group data to be transferred to the channel. The path group data that is transferred is the portion of the device’s path group data associated with the particular channel path on which the command is executed. The first byte is the path state byte. The following 11 bytes are current Path Group ID for the channel interface.

*Figure 21. Path State byte definition*

Bits	Value	Definition
0-1	00	Reset No Set Path Group ID command has been executed on this channel since the last System Reset, subsystem Power On Reset.
	01	Reserved
	10	Ungrouped A valid Path Group ID for this channel exists, and the addressed device does not belong to any group. This state will exist when this device has been ungrouped or a SPID has been executed on this channel to another device (but never to this device).
	11	Grouped A valid Path Group ID exists for this device on this channel, and the device is shared by a group.
2-3	00	The addressed device is not currently reserved.
	01	The addressed device is not currently reserved, but there is a contingent or implicit allegiance to another member of the channel group (when a group exists).
	10	The device is reserved, but not to this channel.
	11	The addressed device is reserved to this channel and to other members of the channel group when one exists.
4	0	Single Path Mode
	1	Multi-path Mode
5-7	Must be zero	

Bytes 1 through byte 11 contain the current Path Group ID that is associated with the addressed device. If no path-group ID command was issued on this interface for any device on this storage subsystem since the last system reset, 11 bytes of zeros will occur in this field. If a valid ID exists for the interface, it will be returned in this field.

## Suspend Multi-path Reconnection

The Suspend Multi-path Reconnection command is intended for failure recovery in multi-path mode. The purpose of the command is to provide the ability to restrict a single channel program to operate on only the path over which the command is issued without having to restructure a path group that already exists.

Suspend Multi-path Reconnection does not reset the first-in-chain internal microcode indication.

Suspend Multi-path Reconnection is not valid in the domain of a Locate Record command; if Suspend Multi-path Reconnection is received within the domain of a

Locate Record, execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the command is accepted, the channel program is restricted to the channel path over which the command was received until final status for the chain is presented and accepted by the channel. If the command is received over a channel path that is in single-path mode, or in multi-path mode with the device in an ungrouped state, the command will have no effect (since the operation has an implied single path). This command does not affect either the state of device reservations or the state of the path groups.

From the time that command is accepted until the chained command, if any, is received, the control unit will return busy status to any Device Reserve command that is received on any other path group for this device. The purpose of this action is to prevent having to present Unit Check busy status to a Device Reserve command that may be chained to the Suspend Multi-path Reconnection. If the chained command is a Device Reserve, then it will be accepted; if the chained command is any other command, then the control unit will begin allowing reservations for this device.

**Note:** On the ESCON interface, if no command is chained to the Suspend Multi-path Reconnection, then the control unit will not prevent reservations for this device.

---

## Subsystem commands

The commands in this section are associated with the subsystem. These commands request the Storage Director to perform actions that may have little or nothing to do with the device.

See Figure 5 on page 19 for a complete list of channel commands.

## Set Subsystem Mode

The Set Subsystem Mode command enables and disables the following hardware or functions:

- Cache for Device
- Cache for Subsystem
- Cache Fast Write Data for Subsystem
- DASD Fast Write for Device
- non-volatile storage for Subsystem

The Set Subsystem Mode command must be the first command in a CCW chain or immediately preceded by a Suspend Multipath Reconnection command that is the first command in the chain. If this condition is not met, the Set Subsystem Mode is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Unless Message Required flag (byte 0, bit 7) is '1', the Set Subsystem Mode command must be the only command in the chain. If Message Required flag is a '1', a Read Message ID can be chained to the Set Subsystem Mode command. If Message Required flag = '0' and any command is chained from the Set Subsystem

Mode command, or if Message Required flag = '1' and any command other than a Read Message ID is chained from the Set Subsystem Mode command, that command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence), but this rejection has no effect on the execution of the Set Subsystem Mode command.

**Note:** If Message Required flag is set but the Set Subsystem Mode command is not followed by a Read Message ID, the operation will be performed; however, notification of completion will not occur.

Following presentation of initial status, two bytes, called the operation field, are transferred from the channel to the Storage Director. If the channel attempts to transfer more than the required bytes, only the required bytes are accepted. If the channel transfers fewer than the required bytes, the command is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required). The two-byte operation field indicates the function to be performed and has the following format:

**B0-B2 Cache Storage Control**

- 0 0 0** No Cache Storage Operation
- 0 0 1** Activate Caching for Addressed Device. Accepted and ignored by ESS.
- 0 1 0** Deactivate Caching for Addressed Device. Accepted and ignored by ESS.
- 0 1 1** Make Cache Storage Available to Subsystem. Accepted and ignored by ESS.
- 1 0 0** Make Cache Storage Unavailable to Subsystem. Accepted and ignored by ESS.
- 1 0 1** Force Cache Storage Unavailable to Subsystem. Accepted and ignored by ESS.
- 1 1 0** Reserved
- 1 1 1** Reserved

**B3-B5 Cache Fast Write Data Control**

- 0 0 0** No Cache Fast Write Data Operation
- 0 0 1** Reserved
- 0 1 0** Reserved
- 0 1 1** Activate Cache Fast Write Access for Subsystem
- 1 0 0** Deactivate Cache Fast Write Access for Subsystem
- 1 0 1** Reserved
- 1 1 0** Reserved
- 1 1 1** Reserved

**B6 (Must be zero)**

**B7 Message Required**

- 0** No message
- 1** Send message when operation complete

**B8-B10 non-volatile storage Control**

- 0 0 0** No non-volatile storage Operation.
- 0 0 1** Activate DASD Fast Write for Addressed Device. Accepted and ignored by ESS.
- 0 1 0** Deactivate DASD Fast Write for Addressed Device. Accepted and ignored by ESS.
- 0 1 1** Force Deactivate DASD Fast Write for Addressed Device.
- 1 0 0** Make non-volatile storage Available to Subsystem. Accepted and ignored by ESS.

- 1 0 1 Make non-volatile storage Unavailable to Subsystem. Accepted and ignored by ESS.
- 1 1 0 Reserved
- 1 1 1 Reserved

**B11-B12 Record Mode Control for TPF**

- 0 0 No change in status of record mode.
- 01-10 Reserved and must be zero unless the subsystem is operating in TPF Mode (TPF Mode is set in VPD). In TPF Mode, see *TPF Support RPQs, GA32-0134* for the operating characteristics of record cache with Multi-Path Lock Facility (RPQ 8B0178).
- 1 1 Reserved

**B13-B15 (Must be zero)**

If the Operation field bit 7 (Message Required flag) is a 1, the system is requesting that it be notified when the operation is complete with an attention message. The Storage Control will generate a Message with a Message Identifier and place it in a buffer that can be read with a Read Message ID. The format of the buffer is the same as the Message that will be returned when the operation is complete. In order to get a Message from the subsystem, the host must both set the Message on and chain a Read Message ID from the Set Subsystem Mode. If either of these two requirements are not met, the subsystem will not generate a Message. If the operation completion status in the message is not equal to '03' (pending), the operation is complete and the subsystem will not owe the system a Message. See "Message Buffer" on page 142 for the format of the Message returned by the subsystem.

The parameters are checked for validity as follows:

- Bit 6 and bits 13 through 15 of the operation field (bytes 0-1) must be zero.
- Unless TPF Mode is set in VPD, bits 11 and 12 of the operation field (bytes 0-1) must be zero.
- The Message Required flag (byte 0, bit 7) is only valid with operations that may take a long time to complete and can continue operation following Device End (Bits 0-2 equal 010, 011 or 100; bits 3-5 equal 100; or bits 8-10 equal 010, 100 or 101).
- Only one control field (Cache Storage, Cache Fast Write Data, Record Mode Control or non-volatile storage) can have an operation specified. The other fields must be '000'.

If these conditions are not met, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If the subsystem is unable to execute the command because of lack of resources, the Storage Control will Channel Command Retry the command until the resources necessary are available. In some cases, space in the message buffer may not be able to be freed. If a response message is required and space cannot be made available, the command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 2B).

All 'Reserved' bit combinations will result in 'No Operation.'

If the condition requested by the operation already exists, the order will perform no function. Depending on the order, successful completion will be indicated with clean ending status or successful completion in the Message.

If the parameter byte contains any other valid operation code, the operation proceeds as follows:

**Note:** All of the conditions set by this command are carried across subsystem IML. All status is kept on the Partition Status Area of the devices chosen for storage of Global status.

### **Activate Cache Fast Write Access for Subsystem**

This subcommand sets indicators to allow Cache Fast Write Access for all devices in the subsystem.

If a Perform Subsystem Function Set Special Intercept Condition was executed for the addressed device on any interface with the same path group identifier as the interface processing the command, processing of this command is terminated with status that includes unit check (Command Reject, Environmental Data Present, Format X'0F' Reason Code X'80') with a program action code of X'70'. See "Set Special Intercept Condition (Order X'1B)" on page 118.

If Cache Fast Write Access is in pending status, this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 04).

Following completion of this command, writes specifying Cache Fast Write Data and containing the current CFWID in the Define Extent command will be allowed to update the cache copy.

This command will be accepted and set the Cache Fast Write Access status to 'active' whether or not the Cache Storage is available to the subsystem. When the Cache Storage becomes available to the subsystem, Cache Fast Write Access will be allowed. This condition will be retained across subsystem IML.

### **Deactivate Cache Fast Write Access for Subsystem**

This subcommand causes Cache Fast Write Access to become inactive for the subsystem and schedules the destage of all modified Cache Fast Write Data in Cache Storage.

If a Perform Subsystem Function, Set Special Intercept Condition was executed for the addressed device on any interface with the same path group identifier as the interface processing the command, processing of this command is terminated with status that includes unit check (Command Reject, Environmental Data Present, Format X'0F' Reason Code X'80') with a program action code of X'70'. See "Set Special Intercept Condition (Order X'1B)" on page 118.

If Cache Fast Write Access is already inactive, Device End is presented with Channel End. If Message Required was set and a Read Message ID command is chained from the SSM command, the Message will indicate successful completion.

Subsequent operations will continue to check the CFWID to insure that it is valid if the Cache Fast Write Data flag is specified but otherwise they will operate as if Cache Fast Write Data had not been specified.



Cache Fast Write Access can be disabled for the subsystem, whether or not the Cache Storage is available to the subsystem or device. Even if the Cache Storage becomes available to the subsystem and caching is active for the device, Cache Fast Write Access will not be allowed for the device unless a Set Subsystem Mode command has allowed Cache Fast Write Access for the subsystem. This condition will be carried across IML.

### **Force Deactivate DASD Fast Write for Device**

This is a recovery subcommand to allow freeing non-volatile storage and Cache Storage allocations to a DASD Fast Write volume when the data cannot be destaged.

This subcommand will:

1. Invalidate all non-volatile storage records for this device if non-volatile storage is available.
2. Invalidate any modified DASD Fast Write records in Cache Storage for this device.
3. Clear the 'Data in NVS for Volume' for this device.

Device End is presented when this activity is complete.

## **Perform Subsystem Function**

The Perform Subsystem Function command performs operations indicated by the order contained in parameter byte 0.

The Perform Subsystem Function command cannot occur within the domain of a Locate Record. If it is within the domain of a Locate Record command, it is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Certain Perform Subsystem Function orders require a level of authorization and must either be:

1. The first command in a chain, or chained directly from a Suspend Multipath Reconnection command that is first in the chain.
2. Immediately preceded by a Define Extent command with the access authority set to 'Diagnostic' or 'Device Support Authorization'.

The Perform Subsystem Function orders that require this level of authorization are as follows:

- Destage Modified Tracks

If the above sequencing requirements are not met, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Additional chaining requirements, if any, are specified in the description of the specific order.

This command must be accepted without regard for device Equipment Check or Intervention Required status.

The first two bytes are common across all orders and are defined as follows:

### Bytes Definition

- 0 Order
- 1 Flags

### Bits Description

- 0 Message Required when the asynchronous processing associated with this order is complete.
- 1-7 Order specific

The valid orders in byte 0 are:

Byte 0	Order
10	Commit
11	Discard
18	Prepare for Read Subsystem Data
19	Destage Modified Tracks
1B	Set Special Intercept Condition
20	Set Cache Allocation Parameters
21	Suspend/Resume
22	Prepare to Read Lock Data
30	Set Lock State
31	Purge Lock
32	Unlock
33	Connect
34	Disconnect
B0	Set-Interface-Identifier

Byte 0 must contain a valid order. If this condition is not met, and these orders are received by the subsystem, the Perform Subsystem Function is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

A variable number of parameter bytes are transferred from the channel (order dependent). If the channel transfers more bytes than the order requires, only the number of bytes required by the order are accepted. If the channel transfers fewer bytes than the order requires, the command is terminated with status that includes unit check (Command Reject, format X'03', CCW byte count less than required).

Additional exceptions, if any, are specified in the description of the specific order.

When the Message Required flag (byte 1, bit 0) is set, the system is requesting that it be notified when the operation is complete with an attention message. The Storage Control will generate a Message with a Message Identifier. This Message can be read with a Read Message ID command chained directly to this command. If the operation completion status in the message is not equal to '03' (pending), the operation is complete and the subsystem will not owe the system a Message. See "Message Buffer" on page 142 for the format of the Message returned by the subsystem.

**Note:** This bit is not accepted on all Perform Subsystem Function orders.

Validity checks for the remaining parameter bytes are order dependent and are defined in the following sections.

The actions performed depend on the order as follows:

### **Commit (Order X'10')**

This order causes the modified data in cache that is within the extents for the addressed device to be written to DASD. This order must be either:

- the first command in a CCW chain,
- chained directly from a Suspend Multipath Reconnection command that is first in the chain,
- chained directly from a Define Extent command,
- Chained directly from another Commit order.

If it is not, the Perform Subsystem Function command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

This order requires that 14 parameter bytes be transferred. The order-unique parameter bytes are defined as follows:

#### **Bytes Definition**

##### **2-5** Beginning Extent Address

This field specifies the track address, in CCHH format, of the beginning track in a range of tracks processed by this order.

##### **6** Must be zero

##### **7-10** Ending Extent Address

This field specifies the track address, in CCHH format, of the ending track in a range of tracks processed by this order.

##### **11** Must be zero

##### **12-13** CFWID

Modified Cache Fast Write and DASD Fast Write data fields on the tracks specified within the bounds of the beginning and ending extents will be written to DASD.

The CCHH values in bytes 2-5 (Beginning extent) and 7-10 (Ending extent) must be a valid primary track address for the addressed device type and the Ending extent must be greater than or equal to the Beginning extent. See Appendix A, "Device data, geometries, and facilities" on page 219.

The extent addresses must meet the criteria defined above and bits 1-6 in the flag byte must be zero. If this condition is not met, the Perform Subsystem Function command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

The flag byte is defined as follows:

<b>Bit</b>	<b>Definition</b>
<b>0</b>	Message Required
<b>1-6</b>	Not used, must be zero
<b>7</b>	=0, only Cache Fast Write modified data will be destaged. =1, both Cache Fast Write and DASD Fast Write modified data will be destaged.

If this command is preceded by a Define Extent and the specified extents are not within the Define Extent specified extents, the command is terminated with status that includes unit check (File Protected).

If Cache Fast Write is activated, the CFWID bytes must be nonzero and must contain the CFWID of the subsystem. If it is zero, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter). If it is nonzero and not the current subsystem CFWID, the command is terminated with status that includes unit check (Command Reject, format F, message 6, Cache Fast Write Access Not Authorized, Reason Code X'01'). If Cache Fast Write is not activated, the CFWID bytes will not be tested for validity.

If flag bit 7=1 so that DASD Fast Write modified data are also to be destaged by the COMMIT order, additional error handling logic is invoked, where necessary:

- If there are data in a failed non-volatile storage for the addressed device, the command will be terminated with status that includes unit check, Command Reject, format X'0F', Status Not as Required, Reason Code X'06'.

If the chain has only one Commit order, the Message Required flag (byte 1, bit 0) determines if the order is synchronous or asynchronous. If the message required flag is set to '0' the operation is synchronous. If the message required flag is set to '1', the operation is asynchronous.

If multiple Commit orders are in a chain, the message required flag in the first command determines if all the Commit orders in the chain are synchronous or asynchronous. If the message required flag is set to '0' in the first Commit order, the remainder of the chain runs synchronously. If the message required flag is set to '1' in the first Commit order, the remainder of the chain runs asynchronously.

Only a Perform Subsystem Function with a Commit or a Read Message ID command can be chained after this order and a Read Message ID can only be chained if the Message Required flag = 1 for the first order in the chain. If this condition is not met, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

### **Discard (Order X'11')**

This order causes specified data for the addressed device to be invalidated in the cache or optionally the cache and non-volatile storage without writing it to DASD.

**Note:** This command is used to discard Cache Fast Write Data data when it is no longer required. It is also used as a recovery tool to remove 'pinned' data from the non-volatile storage and Cache Storage. Incorrect use of the command can result in loss of data.

This order must be either:

- the first command in a CCW chain,
- chained directly from a Suspend Multipath Reconnection command that is first in the chain,
- chained directly from a Define Extent command, or
- Chained directly from another Discard order.

If it is not, the Perform Subsystem Function command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

This order requires 12 parameter bytes be transferred; the order-unique parameter bytes are defined as follows:

#### **Bytes Definition**

**2-5** Beginning Extent Address

This field specifies the track address, in CCHH format, of the beginning track in a range of tracks processed by this order.

- 6** Must be zero when this command is received from a SYS/390 host. When this command is received from a Peer-to-Peer Remote Copy primary control unit, it contains the device address on which to perform the operation specified.

**7-10** Ending Extent Address

This field specifies the track address, in CCHH format, of the ending track in a range of tracks processed by this order.

- 11** Must be zero

The flag byte is defined as follows:

#### **Bit Definition**

- 0** Message Required (must be zero)

- 1** When on, the discard applies to the entire volume. The beginning and ending extents specified in the parameters must be zero.

When off only those tracks specified between the beginning and ending extents will be eligible to be discarded. The CCHH values in bytes 2-5 (Beginning extent) and 7-10 (Ending extent) must be a valid primary track address for the addressed device type unless the device type is unknown, in which case this check is bypassed. The Ending extent must be greater than or equal to the Beginning extent. See Appendix A, "Device data, geometries, and facilities" on page 219.

- 2** When on, all tracks within the specified area are discarded.

When off, only pinned tracks within the area specified are discarded.

- 3** When on, Cache Fast Write, and DASD Fast Write tracks are discarded (made invalid) from cache and non-volatile storage.

When off, only Cache Fast Write tracks in cache are invalidated.

- 4-7** Must be zero.

The flag byte and the extent addresses must meet the criteria defined above or the Perform Subsystem Function command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If the command is preceded by a Define Extent command and the specified extents are not within the specified extents, the command is terminated with status that includes unit check (File Protected).

If the device has data in the failed NVS and bit 3 of the flag byte is binary '1', the command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 06).

Channel End is presented after the parameters are received and verified. For synchronous operation, Device End is presented when all of the appropriate track slot segments are discarded for the addressed volume.

If any command other than another Discard order is chained from this order, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

Channel End is presented after the parameters are received and verified. Device End is presented when all of the appropriate track slot segments are discarded.

If any command other than another Discard order is chained from this order, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

### **Prepare for Read Subsystem Data (Order X'18')**

This order causes the Storage Control to store data in an internal buffer so that a Read Subsystem Data CCW can fetch it.

This order requires 12 parameter bytes be transferred. The order-unique parameter bytes are defined as follows:

#### **Bytes Definition**

- 2-5** Reserved
- 6** Sub-order defining the data to be placed in the internal buffer.

#### **Value Definition**

- 00** The status of the SDs in the subsystem.
- 01** The performance statistics for the subsystem.
- 02** The current CFWID.
- 03** Read attention message for this path-group for the addressed device.  
If the Message Identifier is zero, the message returned is any message waiting to be sent to the host.  
If the Message Identifier is nonzero, the Storage Control will return the current status of the Message Identifier, if it can find one that matches.  
If there is no message to be built, a "No Message" message is built and returned.
- 05** The CCHH addresses of the next 8 pinned tracks on the addressed device starting at the track specified in bytes 8-11.
- 0B** Status of the Logical Paths
- 20** Read Cache Allocation Data (only valid in TPF Mode— see *TPF Support RPQs, GA32-0134*, for operating characteristics.)
- 7** Dependent on the value in byte 6 as follows:
  - Byte**
  - 6** **Description of byte 7**
  - 00** Must be zero

**01** Scope of Statistics to be Returned

If zero, only the first set of statistics for the addressed device are prepared. If 'FF', all statistics for all devices are prepared. All other values are invalid.

**02-FF** Must be zero

**8-N** Additional Parameters. Unless specified otherwise there are four additional parameter bytes (bytes 8-11).

**Byte**

**6** Description of bytes 8-N

**00** Must be zero

**01** Byte 8 indicates the Performance Statistics field length per device. Bytes 9-11 must be zero.

**8 =** Description

**00** Read Subsystem Data (RSSD) reads 96 byte per device Performance Statistics format.

**01** RSSD reads 192 byte per device Performance Statistics format.

**02** Must be zero

**03** Message Identifier

**04, 07** Must be zero

**05** The CCHH of the track used to initiate the search for pinned tracks

The parameters are checked as follows:

- The parameter byte 6 must contain a valid sub-order.
- The flag byte must be zero.
- If the sub-order is other than '01', byte 7 must be zero. If the suborder is '01', only '00' and 'FF' are valid.
- For suborders other than '01', '03', and '05', bytes 8-11 must be zero.
- For suborder '01', bytes 9-11 must be zero.
- If the suborder is '05', it must contain a valid user address (CCHH) for the device. If the device type is unknown, this check is bypassed and any value is accepted.

If these conditions are not met, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If the suborder is '05', the Cache Storage must be available or pending. If this condition is not met, the command is terminated with status that includes unit check, (Command Reject, format X'0F', Status Not as Required, Reason Code 28).

If the suborder is '05' and the Cache Storage is available or pending, it must contain data for the addressed device if the device is 'Data in Failed NVS'. If this condition is not met, the command is terminated with status that includes unit check, (Command Reject, format X'0F', Status Not as Required, Reason Code 2A).

Only a Read Subsystem Data command may be chained from this order. If any other command is chained from this order, it is terminated with status that includes

unit check (Command Reject, format X'02', Invalid Command Sequence) and the contents of the buffer may be lost. The contents of the buffer may also be lost, if the order is not chained but an error will not be reported.

See "Read Subsystem Data" on page 139 for details on the contents of each message.

Channel End is presented after the parameters are received and verified. Device End is presented when the specified data is assembled. In most cases, Channel End and Device End will be presented together.

### **Destage Modified Tracks (Order X'19')**

This order schedules the destage of all modified tracks for the subsystem from cache and non-volatile storage.

If a Perform Subsystem Function Set Special Intercept Condition was executed for the addressed device on any interface with the same path group identifier as the interface processing the command, processing of this command is terminated with status that includes unit check (Command Reject, Environmental Data Present, Format X'0F' Reason Code X'80') with a program action code of X'70'. See "Set Special Intercept Condition (Order X'1B')."

If the non-volatile storage is 'Failed' this order will be terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 06).

This order only requires two parameter bytes be transferred, there are no order-unique parameters for this order.

Bits 1-7 of the flag byte must be zero or the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If the Message Required flag is zero, any command chained after this order will be terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the Message Required flag is one, any command other than a Read Message ID chained after this order will be terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

### **Set Special Intercept Condition (Order X'1B')**

This order sets a condition in the subsystem to cause the subsystem to reject certain commands for the addressed device with a unit check. These commands are named Global CCWs and are defined below.

The Set Special Intercept Condition only applies to those paths with the same Path Group ID (regardless of the grouping status) for the addressed device on which the order is received, in terms of accepting and executing the next global command. If this order is executed, the next and only the next global CCW received for this address on a path with the specified Path Group ID is terminated with status that includes unit check. The sense data is (Command Reject, Environmental Data Present, Format X'0F' Reason Code X'80') with a program action code of X'70'.

The global commands include:



- Set Subsystem Mode with one of the following options:
  - Force Cache Storage Unavailable to the Subsystem
  - Activate Cache Fast Write Data Access for Subsystem
  - Deactivate Cache Fast Write Data Access for Subsystem
- The following PSF orders:
  - Destage Modified Tracks
  - Purge Lock
  - Connect
  - Disconnect
- Diagnostic Control (all options)

In addition, if a device is operating in this condition for a given channel path and attention is included in status for this device for this path, unit check will be added to that status. This inclusion of unit check will not reset the device to normal operation. The sense message produced will include a Program Action Code, which will be set to X'71' and Environmental Data Present will be set (byte 2, bit 3).

This command must be the first command in the chain or immediately preceded by a Suspend Multipath Reconnection that is first in chain. If not, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If this command is directed to a device on an interface for which a path group has not been established, the command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 29).

This order requires 2 parameter bytes, there are no order unique parameters. The flag byte must be set to zero. If the flag byte is not set to zero, the command is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

Channel End and Device End are returned after the parameters are validated and processing is completed.

System Resets follow normal path group rules in terms of resetting the condition. The execution of a global command, resets the condition for the address. The execution of a Selective Reset, Set Path Group ID, or Suspend Multi-Path Reconnection does not reset the condition.

If any command is chained from this command, it is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

### **Set Cache Allocation Parameters (Order X'20')**

Set Cache Allocation Parameters is only valid when the ESS is operating in TPF Mode (TPF Mode is set in VPD). In TPF Mode, see RPQ 8B0178 in the *TPF Support RPQs, GA32-0134*, for the operating characteristics of this order.

## Suspend/Resume (Order X'21')

This command is accepted for compatibility with TPF. It has no effect on the operation of the subsystem. It retains Suspend/Resume status in volatile memory so that it may be returned in Sense Subsystem Status.

**Note:** Suspend/Resume is only valid when the ESS is operating in TPF Mode (TPF Mode is set in VPD). In TPF Mode, see RPQ 8B0178 in the *TPF Support RPQs, GA32-0134*, for the operating characteristics of this order.

## Prepare to Read Lock Data (Order X'22')

This order causes the storage director to store MPLF lock data in an internal buffer so that a Read Subsystem Data command can fetch it.

**Note:** If the subsystem is operating in TPF Mode (that is, TPF Mode is set in VPD), see *TPF Support RPQs, GA32-0134*, for the operating characteristics of this order with record cache and Multi-Path Lock Facility.

A Perform Subsystem Function command specifying this order may be preceded in the command chain by a Read Subsystem Data command.

This order requires 16 parameter bytes, including the order in byte 0. Figure 22 shows the definition of bytes 1–15. If the number of parameter bytes transferred is fewer than required, the command is terminated with status that includes unit check (Command Reject, format 0, message 3, CCW byte count less than required).

Figure 22 (Page 1 of 2). Parameter bytes for prepare to Read Lock Data Order X'22'

Byte	Bit	Definition
1		Flag byte.
1		Bit 0—Message required (must be zero).
1		Bit 1—Read Lock Data Continuation Pointer. For all suborders other than '31', must be zero.  For suborder '31', may be either '0' or '1'. '0' requests that following the retrieval of all of the sets of lock data requested (value in byte 5), the directory search is continued and the next Multi-Path Lock Name found for the specified Multi-Path Lock Partition is added to the retrieved data. This facility will obtain the next Multi-Path Lock Name in the directory, if it exists, to be used as a starting point for continuing the retrieval of Multi-Path Lock Name data by a following Perform Subsystem Function command, Prepare to Read Subsystem Data order, suborder 31.  '1' inhibits the search for the next Multi-Path Lock Name following the retrieval of the set of Multi-Path Lock Names requested.
1		Bits 2–7—Reserved, this field must be set to zero.
2–4		Reserved, must be zeros.
5		Number of Lock Data sets to be retrieved  If the suborder in byte 6 is '31', this byte must contain the count of the number of Multi-Path Lock Data sets to be located. The valid values are decimal 1–24 (X'01–18'). If the suborder in byte 6 is not '31', this byte must be set to zero.

Figure 22 (Page 2 of 2). Parameter bytes for prepare to Read Lock Data Order X'22'

Byte	Bit	Definition	
6	Suborder defining the data to place in the buffer.		
	<b>Value</b>	<b>Description</b>	
	00–2F	Invalid	
	30	Read Multi-Path Lock User ID data. See “Read Multi-Path Lock User ID Data” on page 148 for a description of the data prepared by this suborder.	
	31	Read Multi-Path Lock data. See “Read Multi-Path Lock Data” on page 149 for a description of the data prepared by this suborder.	
6	32	Read Multi-Path Lock partition data. See “Read MPLF Partition Data” on page 150 for a description of the data prepared by this suborder.	
	33–FF	Invalid	
7	Depends on the value in byte 6:		
	<b>Byte 6</b>	<b>Description of byte 7</b>	
	30–31	Multi-Path Lock Partition number	
7	32	Reserved, must be zero.	
	8– 15	Depends on the value in byte 6:	
		<b>Byte 6</b>	<b>Description of bytes 8–11</b>
30		Reserved, must be zeros.	
8– 15	31	The Multi-Path Lock Name to be located. The name specified can be any eight-byte value. If the specified Multi-Path Lock Name is nonzero, a hash directly to that lock name within the specified Multi-Path Lock Partition is made, and if found, the data for this directory entry is retrieved. If the value in byte 5 is greater than ‘1’, the lock directory search will continue from the previous found directory entry and will attempt to retrieve additional Multi-Path Lock Names for the partition. Note that this ordering does not imply a sorted list. To retrieve all the active Multi-Path Lock Names for a partition, specify the initial Multi-Path Lock Name as all zeros so that the search is begun at the start of the directory.	
	32	Reserved, must be zeros.	

The parameters are checked as follows:

- Byte 1, flag byte – for all suborders in byte 6 other than ‘31’, must be X'00'. For suborder ‘31’, can only be X'00' or X'40'.
- Byte 5 – If the suborder in byte 6 is X'31', value must be nonzero and not greater than 24 (X'18'). For all other suborders, must be X'00'.
- Byte 6 – Must contain a valid suborder.
- Byte 7 – Only an initialized Multi-Path Lock Partition is valid.
- Bytes 8–15 – For suborders other than ‘31’, must be zeros.

If these conditions are not met, the command is terminated with status that includes unit check (Command Reject, format 0, message 4, invalid parameter).

MPLF objects must be available. MPLF objects are available if the specified Multi-Path Lock Partition has been initialized and the cache is in an available status. If these conditions are not met, the command is terminated with status that includes unit check (Command Reject, format 0, message F, status not as required, reason code X'91').

Only a Read Subsystem Data command may be chained from this order. If any other command is chained from this order, it is terminated with status that includes

unit check (Command Reject, format 0, message 2, Invalid Command Sequence) and the contents of the buffer may be lost. The contents of the buffer may also be lost if a Read Subsystem data is not chained, but an error will not be reported.

Channel end and device end are presented together after the parameters are received and verified, and the specified data is assembled.

**Programming Note:** When suborder '31' is specified and many locks are allocated, and either a large set of Multi-Path Lock Names are requested or the Read Lock Data Continuation Pointer is requested, the time to search the lock directory and assemble the requested data can take many seconds of time. Some host environments may need to lengthen the missing interrupt timer intervals to allow successful completion of this command without a timeout.

### Set Lock State (Order X'30')

This order requests that the state of the specified Multi-Path Lock be transformed from the state in which the Multi-Path Lock now exists into the state specified in the order.

**Note:** If the subsystem is operating in TPF Mode (that is, TPF Mode is set in VPD), see RPQ 8B0178 in *TPF Support RPQs, GA32-0134*, for the operating characteristics of this order.

If this order completes without unit check, a Read Subsystem Data command chained to this order can be used to determine the results of this order.

Execution of this order may generate an attention interrupt and an associated attention message. If the subsystem attention message buffer is full at the time this order is received, this order is terminated with status that includes unit check (Command Reject, format 0, message F, status not as required, reason code X'95').

This order requires 16 parameter bytes to be transferred. Figure 23 on page 123 shows the definitions of bytes 1–15. If the number of parameter bytes transferred is fewer than required, the command is terminated with status that includes unit check (Command Reject, format 0, message 3, CCW byte count less than required).

If TPF Mode is not set in VPD:

- If this command is preceded by a Define Extent and the MPLF Partition ID does not match the Partition allowed specified by the Define Extent, this command is terminated with status that includes unit check (Command Reject and File Protected).
- If the partition is invalid or not initialized, this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 4C).
- If the user token is not valid or not defined for this channel interface and Partition, this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 4D).

If TPF Mode is set in VPD:

- The partition ID check with the value in Define Extent is not made.

- The partition being invalid or not initialized and the user token being not valid or not defined results in the request failing with the appropriate message code.

Figure 23 (Page 1 of 2). Parameter bytes for Set Lock State Order X'30'

Byte	Bit	Definition
1		Flag byte.
1		Bit 0–Message required (must be zero).
1		Bit 1–The Queue Option.  If '1', and if the requested state is not granted, the user is to be queued on the waiter list for the specified Multi-Path Lock. A maximum of one request per Multi-Path Lock User ID can be queued. If the request is queued, an attention interrupt will be presented when the requested transformation has been made. The attention interrupt will be presented for the device address on the channel address (or any channel address within the path-group that includes the original channel address) established for the Multi-Path Lock User Token (transposed Multi-Path Lock User ID) by the Connect order.  If the queue depth limit is exceeded, a reason code is returned. For specifics on the return codes, see Figure 40 on page 146.
1		Bit 2–The Contention Option.  If '1', the user requests that, while they are a holder of the lock, the user be signalled via an attention interrupt to show contention for the lock. If contention for the lock occurs:  The attention interrupt will be presented for the device address on the channel address (or any channel address within the path-group that includes the original channel address) established for the Multi-Path Lock User Token (transposed Multi-Path Lock User ID) by the Connect order.
1		Bit 3–The Promote/Demote Option.  If '1' and the Multi-Path Lock is held by the Multi-Path Lock User ID specified in bytes 11–13 within the specified Multi-Path Lock Partition specified in byte 10, this request is processed as a promotion or demotion to the state specified in byte 15 from the held state.  If the requesting user does not hold the Multi-Path Lock, this option is ignored.  If the requested transformation cannot be granted, and the queue option was specified, this request is placed on the top of the waiter queue. If the user was not a holder of the lock, the waiter is placed at the bottom of the waiter queue. Extreme caution must be exercised if the Promote/Demote option is used with the queue option because deadlocks can be generated.
1		Bit 4–7–Reserved, must be set to zero.
2–9		Multi-Path Lock Name. The user-assigned eight byte name of the Multi-Path Lock Name. All values except all binary zeros are valid.
10		Multi-Path Lock Partition. The Multi-Path Lock Partition specified in the Connect order for the specified Multi-Path Lock User ID.
11–13		The Multi-Path Lock User Token. The Multi-Path Lock User Token is returned by the Connect order, and is the Multi-Path Lock User Token of the user for which this action is requested.
14		The Multi-Path Lock Filter Mask. The Multi-Path Lock Filter Mask that is to be used with the requested Multi-Path Lock state specified in byte 15. See "Set Lock State processing" on page 156 for further details on the function of this field. Values greater than X'0F' are not valid.

Figure 23 (Page 2 of 2). Parameter bytes for Set Lock State Order X'30'

Byte	Bit	Definition
15		<p>The Multi-Path Lock State. The requested Multi-Path Lock state. Values greater than X'0F' are not valid. If the value X'00' is specified, the Set Lock State order is processed as follows:</p> <ul style="list-style-type: none"> <li>• If the Multi-Path Lock is held by at least one other user and this user is not a holder, the request is processed as any other request that specified a nonzero state.</li> <li>• If the user holds the Multi-Path Lock in a nonzero state and specified the promote/demote option, the request is processed as a promote/demote request. Note that a mask of other than X'0F' may result in not granting the transformation.</li> <li>• If the Multi-Path Lock does not exist (there are no holders or waiters), no Multi-Path Lock will be established but the request will be completed by presenting channel end, device end, and status modifier, if no exception conditions are encountered.</li> </ul>

If this order is not rejected with a unit check condition, the parameters are validated, and the subsystem will attempt to execute the order as specified.

If a parameter error is detected or if the subsystem cannot execute the order as specified: processing is terminated, a message is prepared (see "Multi-Path Lock Facility Format:" on page 144) that contains a detailed reason code (see Figure 40 on page 146) on why the order cannot be successfully executed, and ending status of channel end and device end are presented.

If no parameter errors are detected and the subsystem can execute the order as specified: a message is prepared that contains lock state transformation data and a reason code indicating successful execution of the order. Ending status of channel end, device end, and status modifier are presented.

- Locate the Multi-Path Lock identified in parameter bytes 2–9. If the specified Multi-Path Lock does not exist, sub-allocate an Multi-Path Lock entry, initialize it with the specified name, and set the Multi-Path Lock state to zero. If a free entry does not exist, generate the return code (X'06'), and terminate processing by presenting channel end and device end.

When an entry becomes available as a result of an Unlock or Purge Lock order, users with requests terminated due to unavailability of entries are signaled via an attention interrupt. The attention interrupt is presented for the device address on the channel address (or any channel address within the path-group that includes the original channel address) established for the Multi-Path Lock User Token (transposed Multi-Path Lock User ID) by the Connect order.

**Note:** If the Multi-Path Lock identified in parameter bytes 2–9 does not exist and the requested Multi-Path Lock State specified in byte 15 is X'00', lock space is not allocated and the processing is completed by presenting channel end, device end, and status modifier.

The message prepared for either a successful or unsuccessful execution of this order can be retrieved by chaining a Read Subsystem Data (X'3E') command directly to this Perform Subsystem Function order.

The granting of the Multi-Path Lock is based on the compatibility matrix shown in “MPLF State transformation tables” on page 159.

The Multi-Path Lock is a first-in first-out mechanism. If there are waiters on the queue, an attempt to grant the Set Lock State request will not be made unless the requestor is already the holder and the promote/demote option is specified.

### Purge Lock (Order X'31')

This order requests that a specific user or all users for the specified Multi-Path Lock are to be purged as either a holder and/or waiter of the specified Multi-Path Lock.

**Note:** If the subsystem is operating in TPF Mode (that is, TPF Mode is set in VPD), see RPQ 8B0178 in *TPF Support RPQs, GA32-0134*, for the operating characteristics of this order.

If the command is completed without unit check, the Read Subsystem Data can be used to determine the results of the order and any reason codes that may be returned.

If a Perform Subsystem Function Set Special Intercept Condition was executed for the addressed device on any interface with the same path group identifier as the interface processing the command, processing of this command is terminated with status that includes unit check (Command Reject, Environmental Data Present, Format X'0F', Reason Code X'80') with a program action code of X'70'. See “Set Special Intercept Condition (Order X'1B')” on page 118.

The Purge Lock order requires authorization. It must be first in chain. If this requirement is not met, the command is terminated with status that includes unit check (Command Reject, format 0, message 2, invalid command sequence).

Execution of this order has the potential to generate an attention interrupt and an associated attention message. If the subsystem attention message buffer is full at the time this order is received, this order is terminated with status that includes unit check (Command Reject, format 0, message F, status not as required, reason code X'95').

This order requires 16 parameter bytes to be transferred. Figure 24 shows the definitions of bytes 1–15. If the number of parameter bytes transferred is fewer than required, the command is terminated with status that includes unit check (Command Reject, format 0, message 3, CCW byte count less than required).

Figure 24 (Page 1 of 2). Parameter bytes for Purge Lock Order X'31'

Bytes	Bit	Definition
1		Flag byte.
1		Bit 0–Message required (must be zero).
1		Bits 1–3–Purge Scope Option. Determines the scope of the Purge operation.
<b>Value Definition</b>		
		<b>000</b> Invalid.
		<b>001</b> Purge all waiters for the specified Multi-Path Lock (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10). If there are no waiters for the specified Multi-Path Lock Name (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10), this specification is ignored. The Multi-Path Lock User ID specified in bytes 11–13 must be binary zero.

Figure 24 (Page 2 of 2). Parameter bytes for Purge Lock Order X'31'

Bytes	Bit	Definition
1	010	Purge all holders for the specified Multi-Path Lock (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10). If there are no holders for the specified Multi-Path Lock Name (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10), this specification is ignored. The Multi-Path Lock User ID specified in bytes 11–13 must be binary zero.
	011	Purge all holders and waiters for the specified Multi-Path Lock Name (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10). If there are no holders for the specified Multi-Path Lock Name (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10), this specification is ignored. The Multi-Path Lock User ID specified in bytes 11–13 must be binary zero.
1	100	Invalid.
	101	Purge the Multi-Path Lock User ID (bytes 11–13) for the specified Multi-Path Lock (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10) from the waiter queue. If the Multi-Path Lock User ID is not on the waiter queue, the specification is ignored.
1	110	Purge the Multi-Path Lock User ID (bytes 11–13) for the specified Multi-Path Lock (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10) from the holder table. If the Multi-Path Lock User ID is not a holder, the specification is ignored.
	111	Purge the Multi-Path Lock User ID (bytes 11–13) for the specified Multi-Path Lock (bytes 2–9) within the specified Multi-Path Lock Partition (byte 10) from both the holder table and waiter queue. The specification is ignored if the Multi-Path Lock User ID is neither a holder or a waiter for the specified Multi-Path Lock Name (byte 2–9) within the specified Multi-Path Lock Partition (byte 10).
1		Bits 4–7—Reserved, must be set to zero.
2–9		The Multi-Path Lock Name. The eight-byte name of the Multi-Path Lock Name.
10		Multi-Path Lock Partition. The Multi-Path Lock Partition specified in the Connect order for the specified Multi-Path Lock User ID.
11–13		The Multi-Path Lock User ID. The Multi-Path Lock User ID established via the Set Lock State order and is the Multi-Path Lock User ID of the user for which this action is requested. The Multi-Path Lock User Token is not valid.
14		Reserved, must be set to zero.
15		Reserved, must be set to zero.

If this order is not rejected with a unit check condition, the parameters are validated, and the subsystem will attempt to execute the order as specified.

If a parameter error is detected or if the subsystem cannot execute the order as specified, processing is terminated, a message is prepared (see “Multi-Path Lock Facility Format:” on page 144), that contains a detailed reason code (see Figure 40 on page 146) on why the order could not be successfully executed, and ending status of channel end and device end are presented.

If no parameter errors are detected and the subsystem can execute the order as specified, a message is prepared that contains lock state transformation data and a reason code indicating successful execution of the order. Ending status of channel end, device end, and status modifier are presented.



The message prepared for either a successful or unsuccessful execution of this order can be retrieved by chaining a Read Subsystem Data (X'3E') command directly to this Perform Subsystem Function order.

If any command other than a Read Subsystem Data is chained from this command, it is terminated with status that includes unit check (Command Reject, format 0, message 2, Invalid Command Sequence).

### Unlock (Order X'32')

This order requests that the Multi-Path Lock state for the specified Multi-Path Lock User ID be set to the unlocked state, zero. The resultant state of the Multi-Path Lock is dependent on the status of other holders of the Multi-Path Lock

**Note:** If the subsystem is operating in TPF Mode (that is, TPF Mode is set in VPD), see RPQ 8B0178 in *TPF Support RPQs, GA32-0134*, for the operating characteristics of this order.

If the command is completed without unit check, the Read Subsystem Data can be used to determine the results of the order and reason codes that may be returned.

Execution of this order has the potential to generate an attention interrupt and an associated attention message. If the subsystem attention Message Buffer is full at the time this order is received, this order is terminated with status that includes unit check (Command Reject, format 0, message F, status not as required, reason code X'95').

This order requires 16 parameter bytes to be transferred. Figure 25 shows definitions of bytes 1–15. If the number of parameter bytes transferred is fewer than required, the command is terminated with status that includes unit check (Command Reject, format 0, message 3, CCW byte count less than required).

If TPF Mode is not set in VPD:

- If this command is preceded by a Define Extent and the MPLF Partition ID does not match the Partition allowed that was specified by the Define Extent, this command is terminated with status that includes unit check (Command Reject and File Protected).
- If the partition is invalid or not initialized, this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 4C).
- If the user token is not valid or not defined for this channel interface and Partition, this command is terminated with status that includes unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 4D).

If TPF Mode is set in VPD:

- The partition ID check with the value in Define Extent is not made.
- The partition being invalid or not initialized, and the user token being not valid or not defined results in the request failing with the appropriate message code.

Figure 25 (Page 1 of 2). Parameter bytes for Unlock Order X'32'

Bytes	Definition
1	Flag byte. Must be set to zero.
2–9	The Multi-Path Lock Name. The eight-byte name of the Multi-Path Lock.

Figure 25 (Page 2 of 2). Parameter bytes for Unlock Order X'32'

Bytes	Definition
10	Multi-Path Lock Partition. The Multi-Path Lock Partition specified in the Connect order for the specified Multi-Path Lock User ID.
11–13	The Multi-Path Lock User Token. The Multi-Path Lock User Token returned by the Connect order and is the Multi-Path Lock User ID of the user for which this action is requested.
14–15	Reserved, must be set to zero.

If this order is not rejected with a unit check condition, the parameters are validated, and the subsystem will attempt to execute the order as specified.

If a parameter error is detected or if the subsystem cannot execute the order as specified, processing is terminated, a message is prepared (see “Multi-Path Lock Facility Format:” on page 144), that contains a detailed reason code (see Figure 40 on page 146) on why the order cannot be successfully executed, and that presents ending status of ‘channel end’ and ‘device end’.

If no parameter errors are detected and the subsystem can execute the order as specified: a message is prepared that contains lock state transformation data and a reason code indicating successful execution of the order and that presents ending status of ‘channel end’, ‘device end’, and ‘status modifier’.

The message prepared for either a successful or unsuccessful execution of this order can be retrieved by chaining a Read Subsystem Data (X'3E') command directly to this Perform Subsystem Function order.

### Connect (Order X'33')

This order causes a specific Multi-Path Lock User ID to be connected to the MPLF function for the specified Multi-Path Lock Partition.

**Note:** If the subsystem is operating in TPF Mode (that is, TPF Mode is set in VPD), see RPQ 8B0178 in *TPF Support RPQs, GA32-0134*, for the operating characteristics of this order.

The device address and the channel path on which this order is received and executed determines the device address and channel path (or path group) on which all attention interrupts for this Multi-Path Lock User ID will be raised. If this order completes without unit check, the Read Subsystem Data command can be used to determine the results of this order and to retrieve a transposed Multi-Path Lock User ID called an Multi-Path Lock token. The Multi-Path Lock User Token must be specified in the parameter data for the Set Lock State and Unlock orders.

The first successful Connect order execution against a specific Multi-Path Lock Partition makes that partition active. An active Multi-Path Lock Partition only becomes inactive by executing a Disconnect order with the Global Option or by the Cache Storage becoming unavailable.

If a Perform Subsystem Function Set Special Intercept Condition was executed for the addressed device on any interface with the same path group identifier as the interface processing the command, processing of this command is terminated with status that includes unit check (Command Reject, Environmental Data Present, Format X'0F' Reason Code X'80') with a program action code of X'70'. See “Set Special Intercept Condition (Order X'1B')” on page 118.

The specific functions provided by this order are as follows:

- Connect the specified user as a MPLF user using the specified Multi-Path Lock User ID (bytes 3–5) within the specified Multi-Path Lock Partition ID (byte 2) or Multi-Path Lock Partition Name (bytes 10–29). This function also establishes the address on which all attention interrupts for this Multi-Path Lock User ID will be raised.
- Transpose the specified Multi-Path Lock User ID into an authorized Multi-Path Lock User ID (Multi-Path Lock User Token) that must be used in the Set Lock State and Unlock orders.
- Assume ownership of MPLF objects owned by another user. This function will also change the address on which all attention interrupts for this Multi-Path Lock User ID will be raised.
- Make the Multi-Path Lock User Token, created from the Multi-Path Lock User ID, available for retrieval by the Read Subsystem Data command.

An IML of the subsystem resets all Multi-Path Lock User IDs and all Multi-Path Locks.

The Connect order requires authorization. It must be first in chain. If this requirement is not met, the command is terminated with status that includes unit check (Command Reject, format 0, message 2, invalid command sequence).

This order requires 10 or 30 parameter bytes to be transferred, depending on the value of flag byte bit 4. Figure 26 shows definitions of bytes 1–29. If the number of parameter bytes transferred is fewer than required, the command is terminated with status that includes unit check (Command Reject, format 0, message 3, CCW byte count less than required).

Figure 26 (Page 1 of 2). Parameter bytes for Connect Order X'33'

Byte	Bit	Definition
1		The flag byte.
1	0	Message Required (must be zero).
1	1–2	Function option. The function options are as follows:
		<b>Value Description</b>
	00	Basic connect. The Multi-Path Lock User ID specified in bytes 3–5 is connected to the Multi-Path Lock Partition specified in byte 2 as a MPLF user.
	01	Assume ownership. The attention interrupt address associated with the Multi-Path Lock User ID (bytes 3–5) within Multi-Path Lock Partition (byte 2) is changed to the address on which this order is received.
	10	Read Multi-Path Lock User Token. The Multi-Path Lock User Token for the Multi-Path Lock User ID (bytes 3–5) within Multi-Path Lock Partition (byte 2) is made available for retrieval via a Read Subsystem Data command.
	11	Invalid.

Figure 26 (Page 2 of 2). Parameter bytes for Connect Order X'33'

Byte	Bit	Definition
1	Bit 3	<p>Lock Space Option. If this field is set to zero, use the value specified in Lock Count (bytes 6–9) for lock space allocation. If this field is set to one, the Lock Count field is ignored and the processing is dependent on the function option (bits 1–2) of this byte.</p> <ul style="list-style-type: none"> <li>If the function option in bits 1–2 is '01' or '10', the lock count value established for the Multi-Path Lock Partition is used.</li> <li>If the function option in bits 1–2 is '00' and this is the first Connect for the specified Multi-Path Lock Partition, the maximum number of locks (16k) is allocated to the specified partition. If the Multi-Path Lock Partition was previously initialized, the value established for the Multi-Path Lock Partition is used.</li> </ul>
1	Bit 4	<p>Use Multi-Path Lock Partition Name. If this bit is set, the order requires 30 bytes of data; the last twenty bytes specify a partition name. When this is specified, the system is requesting that the storage facility find or allocate a lock partition for this partition name. If specified, execution proceeds as follows:</p> <ol style="list-style-type: none"> <li>All active partitions are examined to determine if the partition name specified already has an active partition. If one is found, the Multi-Path Lock Partition ID is set to the partition that matches the partition names specified.</li> <li>If no active partition has a name to match the one specified, then an inactive partition is selected, the name specified is assigned to this partition, and the Multi-Path Lock Partition ID is set to the partition selected.</li> </ol> <p><b>Note:</b> If TPF mode is specified in VPD, partitions 0-1 are reserved for TPF usage, leaving only 6 for other use.</p> <ol style="list-style-type: none"> <li>If all partitions are active, the command fails with reason code X'0D' (see "Multi-Path Lock Facility Format:" on page 144).</li> </ol> <p>The data returned by Read Subsystem Data will indicate the Partition ID assigned.</p>
1	Bits 5–7	Reserved, must be set to zero.
2		Multi-Path Lock Partition. The Multi-Path Lock Partition to which the user is to be connected. If flag byte bit 4 is a one (use partition name), this byte must be zero. Otherwise, the valid Multi-Path Lock Partition values are decimal 1–8 (X'01–08').
3–5		The Multi-Path Lock User ID. The Multi-Path Lock User ID to which the user is to be connected. If the Ownership option is specified, the Multi-Path Lock User ID within Multi-Path Lock Partition must equal an Multi-Path Lock User ID within Multi-Path Lock Partition that was connected via a previously executed Connect order. The valid Multi-Path Lock User IDs are decimal 1–32 (X'01–20'). An Multi-Path Lock User Token is not valid.
6–9		Lock Count. The maximum number of Multi-Path Locks that will be concurrently defined for this Multi-Path Lock Partition. All users within an Multi-Path Lock Partition must specify the same Multi-Path Lock count. If the Lock Space option is specified as one, this field is ignored.
10–29		Multi-Path Lock Partition Name These bytes are only transferred if bit 4 of the flag byte is set. See the definition of bit 4 of the flag byte above for the use of this field.

If this order is not rejected with a unit check condition, the parameters are validated, and the subsystem will attempt to execute the order as specified.

If a parameter error is detected or if the subsystem cannot execute the order as specified, processing is terminated, a message is prepared (see "Messages Defined by Sense" on page 210), that contains a detailed reason code (see

Figure 40 on page 146) on why the order could not be successfully executed, and ending status of channel end and device end are presented.

If no parameter errors are detected and the subsystem can execute the order as specified, a message is prepared that contains the Multi-Path Lock User Token and a reason code indicating successful execution of the order, and ending status of channel end, device end, and status modifier are presented.

The message prepared for either a successful or unsuccessful execution of this order can be retrieved by chaining a Read Subsystem Data (X'3E') command directly to this Perform Subsystem Function order.

If any command other than a Read Subsystem Data is chained from this command, it is terminated with status that includes unit check (Command Reject, format 0, message 2, Invalid Command Sequence).

### **Disconnect (Order X'34')**

This order causes a specific user or all users within the specified Multi-Path Lock Partition to be disconnected from the MPLF facility.

**Note:** If the subsystem is operating in TPF Mode (that is, TPF Mode is set in VPD), see RPQ 8B0178 in *TPF Support RPQs, GA32-0134*, for the operating characteristics of this order.

Disconnection results in the deletion of ownership of MPLF objects for the specified user or users within the specified Multi-Path Lock Partition, regardless of their current usage of those objects.

Disconnecting all Multi-Path Lock User IDs by use of individual user disconnect commands (flag Byte Bit 1=0) does not deactivate the Multi-Path Lock Partition. The Global Disconnect option (flag byte bit set to '1') must be used to make the Multi-Path Lock Partition inactive, the state required to either change the Multi-Path Lock Partition size.

If a Perform Subsystem Function Set Special Intercept Condition was executed for the addressed device on any interface with the same path group identifier as the interface processing the command, processing of this command is terminated with status that includes unit check (Command Reject, Environmental Data Present, Format X'0F' Reason Code X'80') with a program action code of X'70'. See "Set Special Intercept Condition (Order X'1B')" on page 118.

The specific functions provided by this order are as follows:

- Disconnect the specified user as a MPLF user and delete this user's ownership of Multi-Path Locks regardless of whether this user is a holder or a waiter for those Multi-Path Locks. As a result of this deletion of ownership, a given Multi-Path Lock will be made available to other users. If the disconnected user was a holder of the object, calculate a new Multi-Path Lock state. If there are waiters, assign the Multi-Path Lock to the top users. If there are no other holders or users, make the Multi-Path Lock available for reuse.
- Disconnect all MPLF users and associated objects within the specified Multi-Path Lock Partition. All Multi-Path Locks within the Multi-Path Lock Partition are made available for reuse.

The Disconnect order requires authorization. It must be first in chain. If this requirement is not met, the command is terminated with status that includes unit check (Command Reject, format 0, message 2, Invalid Command Sequence).

Execution of this order specifying the Single User Disconnect option (flag byte bit 1 set to '1') has the potential of generating an attention interrupt and an associated attention message. If the subsystem attention message buffer is full at the time the Single User Disconnect option of this order is received, this order is terminated with status that includes unit check (Command Reject, format 0, message F, status not as required, reason code X'95').

This order requires 6 parameter bytes to be transferred. Figure 27 shows definitions of bytes 1–5. If the number of parameter bytes transferred is fewer than required, the command is terminated with status that includes unit check (Command Reject, format 0, message 3, CCW byte count less than required.)

Figure 27. Parameter bytes for Disconnect Order X'34'

Byte	Bit	Definition
1		Flag byte.
1		Bit 0—Message Required (must be zero).
1		Bit 1—Global Disconnect option. If '1', all users of the MPLF facility within the specified Multi-Path Lock Partition (byte 2) are disconnected. If '0', only the user identified by the Multi-Path Lock User ID (bytes 3–5) within the specified Multi-Path Lock Partition is disconnected.
1		Bits 2–7—Reserved, must be set to zero.
2		Multi-Path Lock Partition. The Multi-Path Lock Partition for which the user is to be disconnected. The valid Multi-Path Lock Partition values are decimal 1–8 (X'01–08').
3–5		The Multi-Path Lock User ID. If the Global Disconnect option (flag byte bit 1) is set to '0', this field is required. If specified, the Multi-Path Lock User ID must be a valid User ID. The specified Multi-Path Lock User ID cannot be an Multi-Path Lock User Token. If the Global Disconnect option (flag byte bit 1) is set to '1', this field must be set to zero.

If this order is not rejected with a unit check condition, the parameters are validated, and the subsystem will attempt to execute the order as specified.

If a parameter error is detected or if the subsystem cannot execute the order as specified, processing is terminated, a message is prepared (see “Multi-Path Lock Facility Format:” on page 144) that contains a detailed reason code (see Figure 40 on page 146) on why the order could not be successfully executed, and ending status of channel end and device end are presented.

If no parameter errors are detected and the subsystem can execute the order as specified, a message is prepared that contains a reason code indicating successful execution of the order. Ending status of channel end, device end, and status modifier are presented.

- The following actions are asynchronously executed after ending status is presented for a user disconnect. Any MPLF order received for this partition while this action is underway will receive a return code of X'E0'. For each Multi-Path Lock within the specified Multi-Path Lock Partition for which this Multi-Path Lock User ID is a holder or a waiter do the following:

1. Purge the Multi-Path Lock User ID from the waiter and holder queues.
2. If the Multi-Path Lock User ID was a holder, calculate a new Multi-Path Lock state.
3. If the resultant Multi-Path Lock state is zero and there are no entries on the waiter queue, make the Multi-Path Lock available for reuse.
4. If there are waiters on the waiter queue, attempt to grant the Multi-Path Lock to the top user(s) on the wait queue. If the lock was granted to another Multi-Path Lock User ID, present an attention interrupt for that Multi-Path Lock User ID.

The message prepared for either a successful or unsuccessful execution of this order can be retrieved by chaining a Read Subsystem Data (X'3E') command directly to this Perform Subsystem Function order.

If any command other than a Read Subsystem Data is chained from this command, it is terminated with status that includes unit check (Command Reject, format 0, message 2, Invalid Command Sequence).

### **Set Interface Identifier (Order X'B0')**

This order can set an interface identifier for the addressed device and prepares "node-identifier" information to be read by the immediately following Read Subsystem Data command.

This order will be executed even if the addressed device is not installed or not ready.

Only a Read Subsystem Data command may be chained from this order. If any other command is chained from this order, is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

This order requires four parameter bytes be transferred, the order unique-parameter bytes, bytes 2-3, contain the interface ID of an interface for which node-identification information is requested.

If fewer than four parameter bytes are received, the order is terminated with unit check (Command Reject, format 0, message 3, CCW byte count less than required).

Bits 6-7 of the flag byte are the "Node Selector" and identify the type of node descriptor of the node to be identified in the following Read Subsystem Data response, as follows:

#### **Bits**

#### **6-7 Definition**

- |           |  |
|-----------|--|
| <b>00</b> | The node determined by the interface that is processing the Set Interface Identifier order is to be identified by return of its Node Descriptor and Node Qualifier(s) record. The contents of the interface-ID field (bytes 2-3) have no meaning in this case and are ignored. |
| <b>01</b> | The node determined by the interface associated with the specified (bytes 2-3) interface-ID is to be identified, and its Node Descriptor record returned.  |

**10-11** Reserved.

If the flag byte, bits 0-5 are not zero, or if bits 6-7 is binary '11 or 10', the order is terminated with status that includes unit check (Command Reject, format X'04', Invalid Parameter).

If flag byte bits 6-7 equal '01' and bytes 2-3 do not contain a valid interface ID, the order is terminated with unit check (Command Reject, format X'04', Invalid Parameter). If, in addition, bytes 2-3 specify an interface ID that is not valid on the receiving subsystem, the order is terminated with unit check (Command Reject, format X'0F', Status Not as Required, Reason Code 34).

The Set Interface Identifier order can be accepted on parallel or ESCON interfaces. See "Node Descriptor Record" on page 151 for details on the data returned.

When the parameters are validated Channel End and Device End are presented.

## Sense Subsystem Status

The Sense Subsystem Status command causes the transfer to the channel of information describing the status of the ESS subsystem and the addressed device.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The Sense Subsystem Status command must be the first command in a CCW chain or be chained directly from a Read Device Characteristics, Read Configuration Data, or Suspend Multipath Reconnection command. If it is preceded by a Suspend Multipath Reconnection, the Suspend Multipath Reconnection must be first in the chain. If it is not, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence). If any command is chained from the Sense Subsystem Status command, that command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence), but this rejection has no effect on the execution of the Sense Subsystem Status command.

Following presentation of initial status, the Storage Director transfers 44 (40 if byte 0 bits[4-7]=0) bytes of data to the channel. These bytes have the following format:

### Bytes

<b>0</b>	Format of Sense Subsystem Status
<b>1</b>	Device Unit Address
<b>2</b>	Number of Devices
<b>3</b>	Number of Statistics sets per Device
<b>4</b>	Overall Caching Status
<b>5</b>	Overall non-volatile storage Status
<b>6-7</b>	Length of Performance Statistics
<b>8</b>	Reserved
<b>9</b>	Reserved
<b>10-13</b>	Configured Cache Storage Capacity
<b>14-17</b>	Available Cache Storage Capacity
<b>18-21</b>	Pinned Cache Space
<b>22-25</b>	Offline Cache Storage Capacity
<b>26-27</b>	Addressed Device Status



- 28-31 Configured Nonvolatile Storage Capacity
- 32-35 Pinned Nonvolatile Storage Space
- 36 Device Status - Group 2
- 37 Global Status
- 38-39 SSID

Byte 0, Bits 4-7 = 1

40-43 Reserved. Set to zeroes.

End of Byte 0, Bits 4-7 = 1

This data is interpreted as follows:

**Byte 0, Format of Sense Subsystem Status:**

*Figure 28. Sense Subsystem Status, Byte 0, Format of Sense Subsystem Status*

Bits	Value	Definition
0-3		Must be zero.
4-7	Format	
	0	3990 Models 1, 2, and 3 or Basic Operation Mode.
	1	3990-6 Enhanced Operating Mode 1.
	2-F	Reserved

**Byte 1, Device Unit Address:** The address on which this channel program is executing.

**Byte 2, Number of Devices:** This is the number of configured single access volumes and parallel access volumes minus 1 (0-255 => 1-256 Volumes). Unconfigured and Alias Unit Addresses are not included.

**Byte 3, Number of Statistics Sets per Device:** If TPF mode is not set in VPD then this byte contains X'01'. Otherwise this byte will contain X'04', to support TPF.

**Byte 4, Caching Status:** This byte contains a three-bit field defining the caching status of the subsystem and five bit-significant indicators of conditions associated with caching status of the Storage Director receiving the command.

*Figure 29 (Page 1 of 2). Sense Subsystem Status, Byte 4, Overall Caching Status*

Bits	Value	Definition
0-2	Subsystem Caching Status	
	000	Caching is active (Cache Storage is online).
	001	Reserved
	010	An internal subsystem error has occurred that has caused one Cache Storage in the Storage Controller to become unavailable.
	011 - 111	Reserved
3		Reserved
4		The storage facility is running as a single cluster.

Figure 29 (Page 2 of 2). Sense Subsystem Status, Byte 4, Overall Caching Status

Bits	Value	Definition
5		Must be zero
6		Reserved
7		Cache Fast Write Data is Deactivated

**Byte 5, non-volatile storage Status:** This byte contains a two-bit field defining the subsystem non-volatile storage status.

Figure 30. Sense Subsystem Status, Byte 5, non-volatile storage Status

Bits	Value	Definition
0-1		Subsystem non-volatile storage Status
	00	non-volatile storage is active
	01	Failed, an internal subsystem error occurred that terminated non-volatile storage availability. All data has not been successfully destaged or discarded.
	10	Unavailable, Reserved
	11	Pending, Reserved
2		DASD Fast Write is inhibited because of a subsystem condition.
3		One non-volatile storage is disabled for maintenance
4		At least one non-volatile storage is pending because of an error
5-7		Reserved

**Bytes 6-7, Length of Performance Statistics:** The length, in bytes, of the Performance Statistics data, per device. If X'0000', the length is 96 bytes. If X'00C0', the length is 192 bytes. However, the number of bytes read per device by a Read Subsystem Data command following a Perform Subsystem Function command, Prepare for Read Subsystem Data order, requesting performance data (suborder X'01'), is determined by the setting of byte 8 for the latter suborder. Depending on the setting of this byte, a 96-bytes-per-device field can be read, although the stored format is 192 bytes. See Perform Subsystem Function command, Prepare for Read Subsystem Data order, suborder X'01', for a discussion of this point.

**Note:** The current setting of bytes 6-7 is X'00C0'.

**Byte 8, Reserved.**

**Byte 9, Reserved.**

**Bytes 10-13, Configured Cache Storage Capacity:** This capacity is in bytes if byte 0, bits 4-7 = 0; it is in KB if byte 0 bits 4-7 = 1.

**Bytes 14-17, Available Cache Storage Capacity:** The quantity of Cache Storage that is currently available to the Cluster Processor Complex that controls this subsystem for allocation as cache space. This is the total online space available to this subsystem for caching. This quantity is in bytes if byte 0 bits 4-7 = 0; it is in KB if byte 0 bits 4-7 = 1.

**Bytes 18-21, Pinned Cache Storage Space:** The quantity of Cache Storage that is currently allocated for Pinned data within this Cluster Processor Complex (data that cannot be transferred to DASD because of an error). This quantity is in bytes if byte 0 bits 4-7 = 0; it is in KB if byte 0 bits 4-7 = 1. If byte 0 bits 4-7 = 1 and the

number of bytes pinned is less than 1 KB, the content of this field will be set to one.

**Bytes 22-25, Offline Cache Storage Capacity:** The quantity of Cache Storage that is currently unavailable to the Storage Director for allocation as cache space because of Cache Storage failures that occurred during attempts to read from Cache Storage. This quantity is in bytes if byte 0 bits 4-7 = 0; it is in KB if byte 0 bits 4-7 = 1.

**Programming Note:** The sum of Available Cache Storage Capacity, Pinned Cache Storage Space, and Offline Cache Storage Capacity is always less than the Configured Cache Storage Capacity due to space occupied by subsystem control data.

**Bytes 26-27, Device Status:** This field contains the status for the addressed device of the following functions: Cache, and DASD Fast Write. This status data is also included in the data returned via the Read Subsystem Data prepared by the Perform Subsystem Function Prepare for Read Subsystem Data (suborder 01).: The bits in these bytes are defined as follows:

Figure 31. Sense Subsystem Status, Byte 26, Device Status - Group 1, 1/2

Bits	Value	Definition
0-1		Device Caching Status - must be set to zeros to indicate 'Caching Activated'
2-3		DASD Fast Write Status for Device - must be set to zeros to indicate 'DASD Fast Write Allowed'
4		Reserved
5		Reserved
6-7		Reserved

Byte 0, Bits 4-7 = 0

Figure 32. Sense Subsystem Status, Byte 27, Device Status - Group 1, 2/2

Bits	Value	Definition
0-1		Pinned data
	00	No Pinned data exists for the device
	01	Pinned Data exists for the device
	10	Reserved.
	11	Reserved
2-7		Reserved

End of Byte 0, Bits 4-7 = 0

Byte 0, Bits 4-7 = 1

Figure 33. Sense Subsystem Status, Byte 27, Device Status - Group 1, 2/2

Bits	Value	Definition
0-7	Reserved	

End of Byte 0, Bits 4-7 = 1

**Bytes 28-31, Configured non-volatile storage Capacity:** Configured non-volatile storage capacity available to the Cluster Processor Complex that controls this subsystem. This capacity is in bytes if byte 0 bits 4-7 = 0; it is in KB if byte 0 bits 4-7 = 1.

**Bytes 32-35, Pinned non-volatile storage Space:** The quantity of non-volatile storage that is currently allocated for Pinned data within this Cluster Processor Complex (data that cannot be transferred to DASD because of an error). This quantity is in bytes if byte 0 bits 4-7 = 0; it is in KB if byte 0 bits 4-7 = 1. If byte 0 bits 4-7 = 1 and the number of bytes pinned is less than 1 KB, the content of this field will be set to one. This value is X'FFFFFFFF' if the non-volatile storage is failed and X'00000000' if no tracks are pinned.

**Byte 36, Device Status - Group 2:** This byte contains additional bit significant status for the addressed device. This byte is also included in the data prepared by the Perform Subsystem Function Prepare for Read Subsystem Data order, suborder 01.

The bits in this byte are defined as follows:

Figure 34. Sense Subsystem Status, Byte 36, Device Status - Group 2

Bits	Value	Definition
0	Reserved	
1	Reserved	
2		Data exists in the failed non-volatile storage for this device
3	Reserved	
4	Zero, Reserved	
5		Volume is part of a RAID Rank that is under going RAID rebuild.
6-7	Reserved, when Byte 0 bits 4-7 = 0	
	Pinned data, when Byte 0 bits 4-7 = 1	
	00	No Pinned data exists for the device
	01	Pinned Data exists for the device
	10	Reserved.
	11	Reserved

**Byte 37, Global Status:** This field defines additional status of functions that affect the subsystem.

The bits in this byte are defined as follows:

Figure 35. Sense Subsystem Status, Byte 37, Global Status

Bits	Definition
0	Cache Fast Write and DASD Fast Write functions are suspended. See "Suspend/Resume (Order X'21')" on page 120.
1-7	Reserved.

**Bytes 38-39, SSID:** Subsystem Identifier assigned to this subsystem. If the CCW count is greater than 44 (40 if byte 0 bits[4-7]=0), only 44 (40 if byte 0 bits[4-7]=0) bytes are transferred. If the CCW count is less than 44 (40 if byte 0 bits[4-7]=0), only the specified number of bytes are transferred.: Channel End and Device End are presented when data transfer is complete.

## Read Subsystem Data

This command transfers to the channel the data requested by the Perform Subsystem Function command.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

This command must be chained from a Perform Subsystem Function with a Prepare for Read Subsystem Data or Set-Interface-Identifier order or execution is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

If the CCW count is greater than the number of bytes specified for the requested message, only the number of bytes specified for the requested message are transferred. If the CCW count is less than the number of bytes specified for the requested message, the CCW count determines the number of bytes transferred.

Channel End and Device End are presented when the transfer is complete.

If any command other than a Perform Subsystem Function with a Prepare for Read Subsystem Data order is chained from a Read Subsystem Data command, it is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The format of the data returned is dependent on the preceding Perform Subsystem Function order and in the case of the Prepare for Read Subsystem Data, the sub-order in byte 6. The message type associated with each Prepare for Read Subsystem Data sub-order in byte 6 is:

### Byte 6 Message Type

<b>00</b>	Storage Control Path Status
<b>01</b>	Performance Statistics
<b>02</b>	CFWID
<b>03</b>	Message Buffer
<b>05</b>	Pinned Track Data
<b>0B</b>	Status of the Logical Paths

## 20 Read Cache Allocation Data

**Note:** Reserved except when TPF Mode is set in VPD. In TPF Mode, see RPQ 8B0178 in *TPF Support RPQs, GA32-0134* for operating characteristics.

Details of the data received for each Prepare for Read Subsystem Data sub-order and the "Node Descriptor" returned by a Set-Interface-Identifier follow:

### Storage Control - Path Status

If byte 6 of the Perform Subsystem Function Prepare for Read Subsystem Data is '00', a 24-byte record is read. If 3990 emulation level Basic Operation Mode is set in VPD, only the first 16 bytes are sent.

For ESS, all bytes in this record are provided for compatibility purposes only. The data is set to emulate an "operational" storage path with a full complement of system adapters configured and enabled with one logical path established through each Storage Cluster. This is a "canned" response that is given each time this information is requested. The format of the data returned is:

Figure 36. Path Status

Byte	Definition
0	Status of Storage Cluster 0
0	Set to one
1	Volume attaches though this Cluster Processor Complex.
2	Cluster Processor Complex Disabled
3	Reserved
4	Command received on this Storage Cluster. This bit is set if a SA in Storage Cluster 0 processed this command.
5-7	Reserved
1	Bit map of SAs configured in this Storage Cluster through I/O board 0
2	Bit map of SAs enabled in this Storage Cluster through I/O board 0 A bit off in this bit map implies that the SA is either not configured or is in maintenance mode.
3	Bit map of SAs fenced in this Storage Cluster through I/O board 0
4	Status of Storage Cluster 0
0	Set to one
1	Volume attaches though this Cluster Processor Complex.
2	At least one Storage Controller is disabled in this Cluster Processor Complex
3	Reserved
4	Must be zero
5-7	Reserved
5	Bit map of SAs configured in this Storage Cluster through I/O board 1
6	Bit map of SAs enabled in this Storage Cluster through I/O board 1 A bit off in this bit map implies that the SA is either not configured or is in maintenance mode.
7	Bit map of SAs fenced in this Storage Cluster through I/O board 1
8-15	Status of Storage Cluster 1. This is the same as bytes 0-7, except that it is for Storage Cluster 1.
16-23	Must be zero.

## Performance Statistics

If byte 6 of the Perform Subsystem Function Prepare for Read Subsystem Data is '01', the performance statistics record or records for one or all of the devices attached to the subsystem is read. If TPF mode is set in VPD and if this host did not specify that it supported ESS in a Set System Characteristics then see RPQ 8B0178 in the *TPF Support RPQs, GA32-0134*, for the data returned when Performance statistics are requested. Otherwise, the following description of the data returned represents a single record or set of statistics. If multiple sets are requested via the Perform Subsystem Function Prepare for Read Subsystem Data order, multiple records are returned as a single entity. For example, if the number of records returned is 64, 192 X 64 (12,288) bytes of data are returned, if the 192-byte record size was requested.

**Note:** The number of devices and the length of the record are included in bytes 2 and bytes 6-7 of the data read by the Sense Subsystem Status. If byte 8 of the Prepare for Read Subsystem data order is zero, only 96-byte records will be transferred.

If byte 7 of the Prepare for Read Subsystem data order is zero, and the addressed device is an Alias of a parallel access volume then the counters returned are those of the Base for that parallel access volume. If byte 7 of the Prepare for Read Subsystem data order is 'FF'x, statistics are not reported on unconfigured or Alias addresses.

The records for specific devices can be identified by the 'Device Identification' field (byte 1) in each record. See "Prepare for Read Subsystem Data (Order X'18')" on page 116 for additional details.

The following is a description of the contents of a performance statistics record.

Figure 37 (Page 1 of 2). Performance Statistics

Bytes	Bits	Definition	
0	0	When this bit = 1, it indicates that the Cache Storage is not available and all counts are zero. (Set to zero for ESS)	
	1-3	Format of Performance Data Returned	
		<b>Value</b>	<b>Definition</b>
		000	3990 Format
001	ESS Format		
4-7	Format of data returned. This field has the same definition and values of bits 4-7 of byte 0 from the Sense Subsystem Status command.		
1	Device Unit Address of the device to which the statistics pertain.		
2-3	Device Status		
	The device status data is identical to the data returned by the Sense Subsystem Status in bytes 26-27.		
	<b>Note:</b> The format of the data in these bytes is determined by the value in byte 0, bits 4-7.		
4-7	Search/Read Normal I/O Requests		
8-11	Search/Read Normal I/O Request Hits		
12-15	Write Normal I/O Requests		
16-19	DASD Fast Write I/O Request Hits		
20-23	Search/Read Sequential I/O Requests		
24-27	Search/Read Sequential I/O Request Hits		
28-31	Write Sequential I/O Requests		

Figure 37 (Page 2 of 2). Performance Statistics

Bytes	Bits	Definition
32-35		DASD Fast Write Sequential I/O Request Hits
36-39		Search/Read Cache Fast Write I/O Requests
40-43		Search/Read Cache Fast Write I/O Request Hits
44-47		Cache Fast Write I/O Requests
48-51		Cache Fast Write I/O Request Hits
52-55		Inhibit Cache Loading I/O Requests that operate with DASD (Set to zeros for ESS.)
56-59		Bypass Cache I/O Requests (Set to zeros for ESS.)
60-63		Sequential DASD to Cache Transfer Operations
64-67		DASD to Cache Transfer Operation Count
68-71		Cache to DASD Transfer Operation Count
72-75		DASD Fast Write Operations Delayed Due to non-volatile storage Space Constraints
76-79		Normal 'DASD Fast Write' Write Operation Counts
80-83		Sequential Access 'DASD Fast Write' Write Operation Counts
84-87		Number of record cache Read Misses
88		Device Status - Group 2  The device status data is identical to the data returned by the Sense Subsystem Status in byte 36. See "Sense Subsystem Status" on page 134.  <b>Note:</b> The format of the data in this byte is determined by the value in byte 0, bits 4-7.
89-92		Quick Write Promotes
93		Reserved.
94-95		SSID.
96- 191		Reserved.

## CFWID

If byte 6 of the Perform Subsystem Function Prepare for Read Subsystem Data, is '02', the current Cache Fast Write Data Identifier (CFWID) for the Subsystem is read. The CFWID is a two-byte counter. The value is used in the Define Extent command to assure that Cache Fast Write Access is valid.

## Message Buffer

If byte 6 of the Perform Subsystem Function Prepare for Read Subsystem Data is '03', the Attention message for this path-group is returned. The message is a variable length field of up to 4096 bytes with the following format:

### Byte Description

0-1 Length

2 Format

#### Value Description

- 0 No message
- 1 Reserved
- 2 ESS message
- 3 Reserved
- 4 Multi-Path Lock Facility
- 5-FF Reserved

3 Message code



**4-7** Message ID, if the request was by Message ID, these bytes contain the requested ID, even if 'No Message' is indicated in byte 2.

**8** Flags = 0

**9-FFF** Format and message dependent

***Delayed Response Format:***

*Figure 38. Delayed Response Format*

Bytes	Definition																
<b>0-1</b>	Length = X'000B'																
<b>2</b>	Format = 02																
<b>3</b>	Message Codes  <table border="0"> <thead> <tr> <th>Code</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>00</b></td> <td>Reserved</td> </tr> <tr> <td><b>01</b></td> <td>Delayed Response Message</td> </tr> <tr> <td><b>02-FF</b></td> <td>Reserved</td> </tr> </tbody> </table>	Code	Description	<b>00</b>	Reserved	<b>01</b>	Delayed Response Message	<b>02-FF</b>	Reserved								
Code	Description																
<b>00</b>	Reserved																
<b>01</b>	Delayed Response Message																
<b>02-FF</b>	Reserved																
<b>4-7</b>	Message ID - The ID assigned by the subsystem to the requested operation. If the request is completed without any asynchronous activity, this field is zeros.																
<b>8</b>	Flags = 00																
<b>9</b>	Operation Completion Status  <table border="0"> <thead> <tr> <th>Code</th> <th>Condition</th> </tr> </thead> <tbody> <tr> <td><b>00</b></td> <td>Successful: For Destage Modified Tracks all modified data was destaged and the No Modified Data indicator is set.</td> </tr> <tr> <td><b>01</b></td> <td>Completed with errors: The resources necessary to start the operation (scan) are available but portions of the operation failed.</td> </tr> <tr> <td><b>02</b></td> <td>Failed: The resources necessary to start the operation are not available.</td> </tr> <tr> <td><b>03</b></td> <td>Pending: This is returned when the host request information on a specific message and the operation that owes that message is still in progress. When read by the Read Message ID command chained from the order, asynchronous operation is required.</td> </tr> <tr> <td><b>04</b></td> <td>Defective Battery: non-volatile storage made available but Defective Battery.</td> </tr> <tr> <td><b>05</b></td> <td>No Modified Data indication is not set. A Destage Modified Tracks order was executed but write operations were detected which created modified data in cache.</td> </tr> <tr> <td><b>07-FF</b></td> <td>Reserved</td> </tr> </tbody> </table>	Code	Condition	<b>00</b>	Successful: For Destage Modified Tracks all modified data was destaged and the No Modified Data indicator is set.	<b>01</b>	Completed with errors: The resources necessary to start the operation (scan) are available but portions of the operation failed.	<b>02</b>	Failed: The resources necessary to start the operation are not available.	<b>03</b>	Pending: This is returned when the host request information on a specific message and the operation that owes that message is still in progress. When read by the Read Message ID command chained from the order, asynchronous operation is required.	<b>04</b>	Defective Battery: non-volatile storage made available but Defective Battery.	<b>05</b>	No Modified Data indication is not set. A Destage Modified Tracks order was executed but write operations were detected which created modified data in cache.	<b>07-FF</b>	Reserved
Code	Condition																
<b>00</b>	Successful: For Destage Modified Tracks all modified data was destaged and the No Modified Data indicator is set.																
<b>01</b>	Completed with errors: The resources necessary to start the operation (scan) are available but portions of the operation failed.																
<b>02</b>	Failed: The resources necessary to start the operation are not available.																
<b>03</b>	Pending: This is returned when the host request information on a specific message and the operation that owes that message is still in progress. When read by the Read Message ID command chained from the order, asynchronous operation is required.																
<b>04</b>	Defective Battery: non-volatile storage made available but Defective Battery.																
<b>05</b>	No Modified Data indication is not set. A Destage Modified Tracks order was executed but write operations were detected which created modified data in cache.																
<b>07-FF</b>	Reserved																
<b>10</b>	If the message is a response to a 'Make Cache Storage Available' or a 'Make non-volatile storage Available' request, bits 0-3 are a bit map of caching Storage Controls in the subsystem with each bit set indicating the physical SP is caching (Bit 0 = 1 = SP 0 caching). Set to 'F0'x by ESS to emulate 3990.  Must be zero for other orders.																

**Multi-Path Lock Facility Format:** Unique ending status may be presented for this command when using Multi-Path Lock Facility.: On completion of data transfer, channel end and device end are presented. For Multi-Path Lock Facility Perform Subsystem Function orders 30 through 34 that complete without including status modifier in ending status, the ending status for Read Subsystem Data on completion of the transfer is unit exception, channel end and device end.

Figure 39 (Page 1 of 2). Multi-Path Lock Facility Message Format

Byte	Description	Definition applicable for:							
		message type 1, for PSF order:					message type:		
		30	31	32	33	34	2	3	4
0-1	Length = X'001A'								
2	Format = X'04'								
3	The message type. The types are as follows: <b>Code Description</b> <b>00</b> Not used. <b>01</b> In-line response. This message details the results of the immediately preceding Perform Subsystem Function, Multi-Path Lock Facility order (Set Lock State, Purge Lock, Unlock, Connect, or Disconnect). <b>02</b> Contention response. This message reports details associated with a Perform Subsystem Function Set Lock State order requested contention signal. <b>03</b> Lock granted. This message reports details for the granting of a Multi-Path Lock that was queued. <b>04</b> Resources available. A previous MPLF request was terminated because of the lack of resources. This message shows that resources were made available.								
4-8	Reserved, set to zero.								
9	The reason code. See Figure 40 on page 146.	X	X	X	X	X			
10-17	The Multi-Path Lock Name.	X	X	X			X	X	
18	The Perform Subsystem Function order number. Set to X'30' (Set Lock) for Delayed Response Messages (message types 2, 3, 4).	X	X	X	X	X	X	X	X
19	The Multi-Path Lock Partition.	X	X	X	X	X	X	X	X
20-22	The Multi-Path Lock User Token returned in the connect order response message for all values in byte 9, except values X'10' through X'1B', for which this field will contain the Multi-Path Lock User ID of the first holder found in a scan of holders, starting with Multi-Path Lock User ID.	X					X	X	X
	The Multi-Path Lock User ID that was specified in the Purge Lock order parameters. This field is set to zero if the all waiters, all holders, or all waiters and holders scope was specified.			X					
	The Multi-Path Lock User Token returned in the connect order response.				X				
	If the return code is X'00', the Multi-Path Lock User Token is returned. If the return code is not zero, the Multi-Path Lock User ID is returned.					X			
	The Multi-Path Lock User ID that was specified in the Disconnect order parameters. This field will be set to zeros if the Global Disconnect option was specified.						X		
23	The current Multi-Path Lock state.	X	X	X			X	X	
24	The requested Multi-Path Lock state.	X					X	X	

Figure 39 (Page 2 of 2). Multi-Path Lock Facility Message Format

Byte	Description	Definition applicable for:							
		message type 1, for PSF order:					message type:		
		30	31	32	33	34	2	3	4
25	The state that the Multi-Path Lock User ID holds the lock.	X					X	X	

Figure 40 (Page 1 of 2). Multi-Path Lock Facility Message Reason Codes

Reason Code	Description	Applies to Lock Order				
		30	31	32	33	34
X'00'	Normal Completion. All requested actions were taken.	X	X	X	X	X
X'01'	The Multi-Path Lock User ID is zero or greater than 32.		X		X	X
X'02'	A Reserved field is not specified as zero.	X	X	X	X	X
X'03'	The specified lock state and/or mask is not valid. The valid values for both the requested state and mask are decimal 0–15 (X'0–F').	X				
X'04'	A nonzero Multi-Path Lock User ID was specified for: a Disconnect order that requested the Global Option, a Purge Lock order requesting Purge all Waiter(s) and/or Holder(s).		X			X
X'05'	The specified Multi-Path Lock Partition number is not valid.	X	X	X	X	X
X'06'	This Connect request exceeds the maximum number of locks available to a partition (16k), or this Set Lock State request cannot be granted because all locks are in use for this partition.	X			X	
X'07'	The specified Purge Scope option is not valid.		X			
X'08'	The specified Lock Name was not located or is not valid. A name of all binary zeros is not valid. The specified name does not exist. Lock names are created by Set Lock State orders and may be deleted by Unlock, Purge Lock and Disconnect orders.	X	X	X		
X'0A'	The specified Connect count field is zero and the Lock Space Option is not specified.				X	
X'0B'	The specified Connect options are not compatible.				X	
X'0C'	The specified Connect lock count does not equal the number of locks previously established for this Multi-Path Lock Partition.				X	
X'0D'	The Connect cannot be granted because no MPLF partition matches the specified partition name and all MPLF partitions are active.				X	
X'10'	The requested lock state transformation was not granted, the request was queued, and the requestor currently holds the lock in a nonzero state.	X				
X'11'	The requested lock state transformation was not granted, the request was queued, and the requestor currently does not hold the lock in a nonzero state.	X				
X'16'	The requested lock state transformation was not granted, the request has not been queued because the queue limit for this user would be exceeded, and the requestor currently holds the lock in a nonzero state.	X				
X'17'	The requested lock state transformation was not granted, the request has not been queued because the queue limit for this user would be exceeded, and the requestor currently does not hold the lock in a nonzero state.	X				
X'1A'	The requested lock state transformation was not granted, the request has not been queued because the Queue Option was not specified, and the requestor currently holds the lock in a nonzero state.	X				
X'1B'	The requested lock state transformation was not granted, the request has not been queued because the Queue Option was not specified, and the requestor currently does not hold the lock in a nonzero state.	X				
X'30'	For the Unlock and Set Lock orders, the User Token is not connected to the specified Multi-Path Lock Partition. For the Purge Lock and Disconnect orders, the User ID is not connected to the specified Multi-Path Lock Partition. For the Connect order with the Assume Ownership or Read Multi-Path Lock User Token order, the User ID is not connected to the specified Multi-Path Lock Partition.	X	X	X	X	X
X'31'	The Multi-Path Lock User ID specified for this Connect order is currently in use.				X	
X'32'	The specified User Token is invalid for this Multi-Path Lock.	X		X		
X'40'	The requested Disconnect order cannot be executed because the Multi-Path Lock Partition requested is not initialized.					X
X'50'	Unlock request not granted because requesting User does not hold the lock in a nonzero state.			X		
X'E0'	An asynchronous Disconnect operation is in progress for this partition.	X	X	X	X	X

Figure 40 (Page 2 of 2). Multi-Path Lock Facility Message Reason Codes

Reason Code	Description	Applies to Lock Order				
		30	31	32	33	34
X'F0'	The order was not executed because Multi-Path Lock Facility space in the Cache Storage has not been initialized, or Cache Storage is not available.	X	X	X	X	X
X'F1'	Cache Storage was reinitialized since the Connect order was executed, either by an automatic reinitialization or by use of the Diagnostic Reinitialize Subsystem order of the Diagnostic Control command. Issue a Disconnect order with the Global Option for this partition.	X	X	X	X	X
X'FF'	An internal Cache Storage reinitialization has failed. Cache Storage is not available to execute Multi-Path Lock Facility orders	X	X	X	X	X

### Pinned Track Data

If byte 6 of the Perform Subsystem Function Prepare for Read Subsystem Data is '05', Pinned Track data is returned.

The pinned track addresses are in CCCCHHTT format in 8 blocks of 4-bytes where:

CCCCHH is the cylinder and head address of a pinned track.

TT is the type of pinned track as follows:

#### Code Type

- 01 Retriable
- 02 Non-Retriable
- 04 Pinned in NVS, Cache Available
- 08 Cache Copy Defective

The next 8 track addresses, starting with CCCCHH provided in bytes 8-11 of the Perform Subsystem Function command, are read. If less than 8 pinned tracks are found, a block of 'FF' represents the end of valid data.

The message is 32 bytes.

### Logical Path Status

If byte 6 of the Perform Subsystem Function Prepare for Read Subsystem Data is '0B', the status of all logical paths in the storage facility is returned. Sixteen bytes are returned for each logical path. The number of logical paths supported by a storage facility is returned in the General NEQ of the Read Configuration Data command.

Each 16 byte record is defined as:

Figure 41. Logical Path Status Entry

Byte	Bit	Definition	
0	0-3	Logical Path Type	
		<b>Value</b>	
		<b>Definition</b>	
		0	Parallel OEMI Channel Not used by ESS
		1	Logical Path not available The logical path is not available because of the hardware configuration.
		2-3	Reserved
		4	ESCON Logical Path that has been established to this subsystem.
		5	9035 Logical Path (not used by ESS)
		6	Logical Path Not Established The logical path is available via an ESCON port but has no host logical path established.
		7-F	Reserved
	4	Fenced	
	5-7	Must be zero	
1-11		Path Group ID if valid, otherwise zero	
12-13		Dependent on the Logical Path type (LP type) in bits 0-3 of byte 0.	
	<b>LP Type</b>	<b>Definition of bytes 12-13</b>	
	0,4	SA ID (0-7; 10-17)	
	6,9	SA ID of lowest SA in range on which the logical path can be established.	
	<b>Otherwise</b>	Zero	
14-15		Dependent on the Logical Path type (LP type) in bits 0-3 of byte 0.	
	<b>LP type</b>	<b>Definition of bytes 14-15</b>	
	4	Host ESCON Full Link address.	
	6,9	SA ID of highest SA in range that the logical path may be established on.	
	<b>Otherwise</b>	Zero	

### Read Multi-Path Lock User ID Data

If the suborder of the Perform Subsystem Function command 'Prepare to Read Lock Data' is '30', the list of active Multi-Path Lock User IDs for the specified Multi-Path Lock Partition are retrieved via the Read Subsystem Data command.

This response returns a variable length message. The length will be dependent on the number of active Multi-Path Lock User IDs.

The maximum number of User IDs allowed to be active in a Multi-Path Lock Partition for this release of the MPLF feature is 32 Multi-Path Lock User IDs.

The format of the data follows:

Figure 42. Read Multi-Path Lock User ID Data Format

Bytes	Description
0	The message length. The minimum length is 16 bytes and the maximum is 140 bytes.
1-2	The number of Multi-Path Lock User IDs for the Multi-Path Lock Partition specified in byte 11.
3-6	The number of Multi-Path Locks allocated for this Multi-Path Lock Partition.
7-10	The number of Multi-Path Locks that are currently in use.
11	The Multi-Path Lock Partition specified in the Perform Subsystem Function command.
12-15	The first Multi-Path Lock User ID entry that is composed of a one byte device address followed by a three byte Multi-Path Lock User ID. The device address is the address established by the Connect order. The Multi-Path Lock User ID is the value specified in the Connect order.
16-19	The second Multi-Path Lock User ID.
20- 135	The third through the 31st Multi-Path Lock User ID
136- 139	The 32nd Multi-Path Lock User ID.

## Read Multi-Path Lock Data

If the suborder of the Perform Subsystem Function command Prepare to Read Lock Data order is '31', a set of one or more Multi-Path Lock data is retrieved via the Read Subsystem Data command.

This response returns a variable length message. The length is dependent on how many Multi-Path Lock Names were found out of the number of Multi-Path Lock Names requested. This release of the MPLF feature allows a maximum of 24 sets of lock data to be retrieved within one command.

The format of the data follows:

Figure 43. Read Multi-Path Lock Data Format

Byte	Description
0-1	The length, variable. The minimum length is 14 bytes and the maximum is 6398.
2-3	The number of Multi-Path Locks returned in this message. If the value in these bytes is zero, the message ends after byte 13.
4-11	Next Multi-Path Lock Name  If Perform Subsystem Function Prepare to Read Lock Data order suborder '31', flag byte bit 1 = '0': <ul style="list-style-type: none"> <li>This field will be nonzero if a search beyond the number of Multi-Path Locks requested in Perform Subsystem Function parameter byte 5 finds another Multi-Path Lock Name for this partition.</li> <li>This field will be zero if no Multi-Path Lock Names are found for this partition, the full number of Multi-Path Lock Names requested are not found for this partition, or an additional Multi-Path Lock Name beyond the number of Multi-Path Lock Names requested is not found for this partition.</li> </ul> If Perform Subsystem Function Prepare to Read Lock Data order suborder '31', flag byte bit 1 = '1', this field will be zero.
12	The Multi-Path Lock Partition for the following Multi-Path Locks.
13	The format indicator for the Multi-Path Lock data that follows. This field is set to X'01' for this release and it equates to length of 266 bytes for each Multi-Path Lock entry.
14- 279	The entry for the first Multi-Path Lock.
280- 545	The entry for the second Multi-Path Lock.
546- 6131	The entry for the third through the 23rd Multi-Path Lock.
6132- 6397	The entry for the twenty-fourth Multi-Path Lock.

Figure 44. Read Multi-Path Lock Data – Lock Entry Format. The format of the (266) byte entry for a given Multi-Path Lock.

Byte	Bit	Description
0-7		The Multi-Path Lock Name.
8		The Multi-Path Lock state.
9-72		The holders of the lock. The entries are ordered by Multi-Path Lock User ID. Each holder is represented by a 2-byte entry. Bytes 9–10 contain an entry for Multi-Path Lock User ID 1, bytes 11–12 for User ID 2, and so on, up to bytes 71–72 for User ID 32.  <b>Note:</b> The 2-byte field for any unused holder entry is zeros.
	0	The contention option. The contention option as requested by this user.
	1-7	Reserved. This field is set to zero.
	8-15	The Multi-Path Lock state that the user holds.
73		The total number of waiters in the queue.
74-255		The queued waiters for the lock. Each waiter is represented by a six-byte entry. The entries are ordered as they are to be processed (first entry at top of queue to last entry at bottom of queue). The first entry in bytes 74–79 is for the next user to be granted the Multi-Path Lock.  <b>Note:</b> The six-byte entries for any unused waiter positions are padded with zeros.
	0	The contention option as requested by this user.
	1	The promotion/demotion option as requested by this user.
	2-7	Reserved. This field is set to zero.
	8-15	The Multi-Path Lock mask as specified by the requester.
	16-23	The Multi-Path Lock state requested by the user.
	24-47	The Multi-Path Lock User ID. A Multi-Path Lock User ID of zero indicates the end of the list.

## Read MPLF Partition Data

When Perform Subsystem Function, Prepare to Read Lock Data order, suborder Read MPLF Partition Data precedes this command, 256 bytes of data are returned. This data is divided into eight 32-byte fields (one field per partition). Bytes 0-31 are for partition 1; 32-63 are for partition 2, and so on through partition 7; bytes 224-255 are for partition 8.

Figure 45. Read MPLF Partition Data

Byte	Bit	Definition
0		Flag
	0	MPLF Partition Active
	1-7	Reserved
1-3		Reserved
4-5		Max number of MPLF locks allocated to this partition
6-7		Number of MPLF locks active for this partition
8-9		Max number of MPLF locks active for this partition since last read of MPLF Partition Data
10-11		Max number of MPLF locks active for this partition since the Partition was initialized.
12-31		MPLF Partition Name



## Node Descriptor Record

If the preceding Perform Subsystem Function command was a Set-Interface-Identifier with a Node Selector = '0', the data returned has the following format:

Figure 46 (Page 1 of 3). Node Descriptor Record

Byte	Description														
<b>Bytes 0–31 contain the Node Descriptor for the Subsystem:</b>															
<b>0</b>	Flags <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>0-2</b></td> <td>Node-ID Validity, set to 0.               <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>0</b></td> <td>Node-ID is valid</td> </tr> <tr> <td><b>1</b></td> <td>Reserved</td> </tr> <tr> <td><b>2</b></td> <td>Node-ID is not valid</td> </tr> </tbody> </table> </td> </tr> <tr> <td><b>3-7</b></td> <td>Reserved</td> </tr> </tbody> </table>	Bit	Description	<b>0-2</b>	Node-ID Validity, set to 0. <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>0</b></td> <td>Node-ID is valid</td> </tr> <tr> <td><b>1</b></td> <td>Reserved</td> </tr> <tr> <td><b>2</b></td> <td>Node-ID is not valid</td> </tr> </tbody> </table>	Bit	Description	<b>0</b>	Node-ID is valid	<b>1</b>	Reserved	<b>2</b>	Node-ID is not valid	<b>3-7</b>	Reserved
Bit	Description														
<b>0-2</b>	Node-ID Validity, set to 0. <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>0</b></td> <td>Node-ID is valid</td> </tr> <tr> <td><b>1</b></td> <td>Reserved</td> </tr> <tr> <td><b>2</b></td> <td>Node-ID is not valid</td> </tr> </tbody> </table>	Bit	Description	<b>0</b>	Node-ID is valid	<b>1</b>	Reserved	<b>2</b>	Node-ID is not valid						
Bit	Description														
<b>0</b>	Node-ID is valid														
<b>1</b>	Reserved														
<b>2</b>	Node-ID is not valid														
<b>3-7</b>	Reserved														
<b>3</b>	Node Type, set to 0 for 'device' type node.														
<b>4-7</b>	Must be zero.														
<b>1-3</b>	Node Parameters <table border="1"> <thead> <tr> <th>Byte</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>Must be zero</td> </tr> <tr> <td><b>2</b></td> <td>Class, set to 1 for DASD</td> </tr> <tr> <td><b>3</b></td> <td>Must be zero</td> </tr> </tbody> </table>	Byte	Description	<b>1</b>	Must be zero	<b>2</b>	Class, set to 1 for DASD	<b>3</b>	Must be zero						
Byte	Description														
<b>1</b>	Must be zero														
<b>2</b>	Class, set to 1 for DASD														
<b>3</b>	Must be zero														
<b>4-9</b>	storage facility Type, in EBCDIC														
<b>10-12</b>	All blanks, in EBCDIC														
<b>13-15</b>	storage facility manufacturer, = "IBM"														
<b>16-17</b>	storage facility manufacturing location, in EBCDIC														
<b>18-29</b>	storage facility sequence number from Storage Cluster zero, in EBCDIC														
<b>30-31</b>	Byte 30 and 31 contain an interface identifier (Tag) that uniquely identifies the physical location of the associated node interface. See "Read Configuration Data" on page 95, bytes 226-227 (General NEQ bytes 2-3) for a definition of these bytes.														
<b>Bytes 32–63 contain the Node Qualifier for the Subsystem:</b>															
<b>32</b>	Flags <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>0-2</b></td> <td>Node Qualifier Contents, set to '0' meaning: The Node-Qualifier field contains a list of interface IDs that are recognized when they are presented using the same interface that was used to request this information.</td> </tr> <tr> <td><b>3-7</b></td> <td>Must be zero.</td> </tr> </tbody> </table>	Bit	Description	<b>0-2</b>	Node Qualifier Contents, set to '0' meaning: The Node-Qualifier field contains a list of interface IDs that are recognized when they are presented using the same interface that was used to request this information.	<b>3-7</b>	Must be zero.								
Bit	Description														
<b>0-2</b>	Node Qualifier Contents, set to '0' meaning: The Node-Qualifier field contains a list of interface IDs that are recognized when they are presented using the same interface that was used to request this information.														
<b>3-7</b>	Must be zero.														
<b>33-39</b>	Must be zero.														

Figure 46 (Page 2 of 3). Node Descriptor Record

Byte	Description														
<b>40-63</b>	A series of one word (four bytes) entries. Each word describes a host interface. The number of valid words is variable since the number of installed interface cards is variable. Any words not required to describe an interface will be set to all zeros. The format for each word is: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>0-1</b></td> <td>Entry Type: indicates the contents of the corresponding word, and is always set to '1' for valid entries, set to '0' for the entries that are not required.                0 - contains no meaningful information                1 - contains an interface ID                2 - contains an interface ID that is the first or last of a range of interface IDs                3 - contains model dependent information</td> </tr> <tr> <td><b>2-3</b></td> <td>Reserved</td> </tr> <tr> <td><b>4-7</b></td> <td>Interface Protocol Type: set to '2' or '6'                0 - unspecified                1 - parallel I/O                2 - ESCON I/O                3 - fiber-extended channel (9034)                4 - fiber-extended control unit (9035)</td> </tr> <tr> <td><b>8-11</b></td> <td>Reserved</td> </tr> <tr> <td><b>12-15</b></td> <td>Subassembly type--set to '1';                0 - unspecified                1 - LED                2 - Laser                3-15: Reserved</td> </tr> <tr> <td><b>16-31</b></td> <td>Interface ID                See "Read Configuration Data" on page 95, bytes 226-227 (General NEQ bytes 2-3) for a definition of these bytes.</td> </tr> </tbody> </table>	Bits	Description	<b>0-1</b>	Entry Type: indicates the contents of the corresponding word, and is always set to '1' for valid entries, set to '0' for the entries that are not required. 0 - contains no meaningful information 1 - contains an interface ID 2 - contains an interface ID that is the first or last of a range of interface IDs 3 - contains model dependent information	<b>2-3</b>	Reserved	<b>4-7</b>	Interface Protocol Type: set to '2' or '6' 0 - unspecified 1 - parallel I/O 2 - ESCON I/O 3 - fiber-extended channel (9034) 4 - fiber-extended control unit (9035)	<b>8-11</b>	Reserved	<b>12-15</b>	Subassembly type--set to '1'; 0 - unspecified 1 - LED 2 - Laser 3-15: Reserved	<b>16-31</b>	Interface ID See "Read Configuration Data" on page 95, bytes 226-227 (General NEQ bytes 2-3) for a definition of these bytes.
Bits	Description														
<b>0-1</b>	Entry Type: indicates the contents of the corresponding word, and is always set to '1' for valid entries, set to '0' for the entries that are not required. 0 - contains no meaningful information 1 - contains an interface ID 2 - contains an interface ID that is the first or last of a range of interface IDs 3 - contains model dependent information														
<b>2-3</b>	Reserved														
<b>4-7</b>	Interface Protocol Type: set to '2' or '6' 0 - unspecified 1 - parallel I/O 2 - ESCON I/O 3 - fiber-extended channel (9034) 4 - fiber-extended control unit (9035)														
<b>8-11</b>	Reserved														
<b>12-15</b>	Subassembly type--set to '1'; 0 - unspecified 1 - LED 2 - Laser 3-15: Reserved														
<b>16-31</b>	Interface ID See "Read Configuration Data" on page 95, bytes 226-227 (General NEQ bytes 2-3) for a definition of these bytes.														
<b>Bytes 64–223:</b>	These bytes contain 5 additional Node Qualifiers for the subsystem. They will contain all zeroes if the interfaces they represent are not installed in ESS. See the definition of bytes 32-63 for a detailed description.														
<b>Bytes 224–415:</b>	These bytes contain additional Node Qualifier's for the Subsystem and contain the link addresses for the interfaces listed in previous Node Qualifier's.														
<b>224</b>	Flags <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>0-2</b></td> <td>Node Qualifier Contents, set to '1' meaning:                The Node-Qualifier field contains a list of link addresses assigned to the port contained in the corresponding interface Node Qualifier's in the previous portion of this command.</td> </tr> <tr> <td><b>3-7</b></td> <td>Must be zero.</td> </tr> </tbody> </table>	Bit	Description	<b>0-2</b>	Node Qualifier Contents, set to '1' meaning: The Node-Qualifier field contains a list of link addresses assigned to the port contained in the corresponding interface Node Qualifier's in the previous portion of this command.	<b>3-7</b>	Must be zero.								
Bit	Description														
<b>0-2</b>	Node Qualifier Contents, set to '1' meaning: The Node-Qualifier field contains a list of link addresses assigned to the port contained in the corresponding interface Node Qualifier's in the previous portion of this command.														
<b>3-7</b>	Must be zero.														
<b>225-231</b>	Must be zero.														

Figure 46 (Page 3 of 3). Node Descriptor Record

Byte	Description								
<b>232-255</b>	A series of one word (four bytes) entries. Each word contains a port link address for a previously described interface. For example, the link address in bytes 232-235 is for the interface id in bytes 42-43. The number of valid words is variable since the number of installed interface cards is variable. Any words not required to describe an interface or for an interface that does not have an address assigned will be set to all zeros. The format for each word is: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>0-1</b></td> <td>Entry Type: indicates the contents of the corresponding word, and is always set to '3' for valid entries, set to '0' for the entries that are not required or for the cases when the link address cannot be determined. 0 - contains no meaningful information 1 - Not Used. 2 - Not Used. 3 - contains a port link address.</td> </tr> <tr> <td><b>2-7</b></td> <td>Reserved</td> </tr> <tr> <td><b>8-31</b></td> <td>Assigned link address. For ESCON, bits 24-31 will contain the link address. For Fibre Channel, bits 8-31 are used. Unused bits are set to zero.</td> </tr> </tbody> </table>	Bits	Description	<b>0-1</b>	Entry Type: indicates the contents of the corresponding word, and is always set to '3' for valid entries, set to '0' for the entries that are not required or for the cases when the link address cannot be determined. 0 - contains no meaningful information 1 - Not Used. 2 - Not Used. 3 - contains a port link address.	<b>2-7</b>	Reserved	<b>8-31</b>	Assigned link address. For ESCON, bits 24-31 will contain the link address. For Fibre Channel, bits 8-31 are used. Unused bits are set to zero.
Bits	Description								
<b>0-1</b>	Entry Type: indicates the contents of the corresponding word, and is always set to '3' for valid entries, set to '0' for the entries that are not required or for the cases when the link address cannot be determined. 0 - contains no meaningful information 1 - Not Used. 2 - Not Used. 3 - contains a port link address.								
<b>2-7</b>	Reserved								
<b>8-31</b>	Assigned link address. For ESCON, bits 24-31 will contain the link address. For Fibre Channel, bits 8-31 are used. Unused bits are set to zero.								

**Bytes 256–415:** These bytes contain additional Node Qualifiers for the subsystem listing the port link addresses. They will contain all zeroes if the interfaces they represent are not installed in ESS. See the definition of bytes 224-255 for a detailed description.

## Read Message ID

This command is used to read the Message Identifier that is assigned by the subsystem to a Set Subsystem Mode or Perform Subsystem Function command that requests notification when an asynchronous operation is complete.

This command is not valid in the domain of a Locate Record command; if it is received in the domain of a Locate Record, it will be rejected with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

The Read Message ID command may only be chained directly from a Set Subsystem Mode or Perform Subsystem Function command that requested a Message. If it is not, the command is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence)

This command must be the last command in the chain. Any command chained from a Read Message ID is terminated with status that includes unit check (Command Reject, format X'02', Invalid Command Sequence).

See "Message Buffer" on page 142 for details on the data returned by this command. The format will be the 11-byte ESS format.

Channel End and Device End are presented when the message transfer is complete.



---

## Chapter 4. Multi-Path Lock Facility

Multi-Path Lock Facility (MPLF) provides function to control processes and share data in a loosely-coupled environment.

**Note:** The ESS provides two support modes for record caching and MPLF. When the ESS is operating in TPF Mode (that is, TPF Mode is set in VPD), see *TPF Support RPQs, GA32-0134*, for the ESS operating characteristics with record cache and Multi-Path Lock Facility.

When the ESS is operating in Basic Operating Mode or in Enhanced Operating Mode 1, the MPLF provides the following capabilities (documented in this manual):

- Establish and maintain a user set of Multi-Path Locks.
- Provide a set of orders (commands) that will conditionally alter the state of the Multi-Path Locks.
- Provide a reason code type response associated with the orders that conditionally alter the state of the Multi-Path Locks. The reason code response shows that the requested state transformation was or was not granted, and identifies any abnormal conditions that may have occurred.
- Provide a set of orders that will provide utilitarian-type functions, such as retrieval and purging of the Multi-Path Locks.

### Multi-Path Lock structure

Multi-Path Locks are named entities that are provided by the MPLF to aid the control of processes and the sharing of data. To minimize the interaction of nonrelated subsystems, the Multi-Path Locks are divided into subsets called Multi-Path Lock Partitions. The Multi-Path Lock Partition acts like a high-level qualifier for both Multi-Path Lock Names and Multi-Path Lock User IDs. As a result, the actions of Multi-Path Lock User ID 'one' and the locks associated with Multi-Path Lock User ID 'one' are unique and separate from those of other users (for example Multi-Path Lock User ID 'two') of a different Multi-Path Lock Partition.

### Multi-Path Lock connect and disconnect

Before using the Multi-Path Lock, a user must establish a Multi-Path Lock User ID within a Multi-Path Lock Partition. This is done via the Perform Subsystem Function–Connect order. This order specifies a Multi-Path Lock User ID and Multi-Path Lock Partition, and the size of the Multi-Path Lock Partition. The maximum size of a Multi-Path Lock Partition is 16k. Having established this connectivity, the user can use Perform Subsystem Function–Set Lock State, –Unlock, and –Purge orders to manipulate the Multi-Path Locks.

When the user completes usage of the MPLF, the user can free suballocated Multi-Path Lock resources by executing a Perform Subsystem Function – Disconnect order. The Disconnect order can be optionally used to disconnect an individually connected Multi-Path Lock User ID from an active Multi-Path Lock Partition, or used to globally disconnect all connected Multi-Path Lock User IDs from an active Multi-Path Lock Partition, thereby de-activating that Multi-Path Lock Partition. The Global Disconnect option must be used for an individual Multi-Path Lock Partition to change the number of Multi-Path Locks allocated for that

Multi-Path Lock Partition or used for all active Multi-Path Lock Partitions to change the total number of Multi-Path Locks initialized in the cache.

## Creating and deleting Multi-Path Locks

Multi-Path Locks are implicitly created and normally deleted based on the execution of Perform Subsystem Function – Set Lock State and Unlock orders, respectively. If a Perform Subsystem Function – Set Lock State order is executed, and there is no Multi-Path Lock with the specified name, a free Multi-Path Lock is assigned this name. If there is no free Multi-Path Lock available, the request is not completed. If a Perform Subsystem Function – Unlock order is successfully executed for a given Multi-Path Lock, that Multi-Path Lock is placed on a free queue if there are no other holders of the Multi-Path Lock. Elements from this queue are available to the Set Lock State order.

Multi-Path Locks can also be made available for reuse via the Purge Lock order.

## Multi-Path Lock state transformations

The state of Multi-Path Locks is transformed by the Perform Subsystem Function – Set Lock State and Unlock orders. The Set Lock State order is normally used to set a Multi-Path Lock to a nonzero or nonfree state and the Unlock is used to set a Multi-Path Lock to the zero or free state.

The number and types of states provided by the MPLF does not necessarily correspond to the classical definition of lock states or Lock Assist states. The user selects a subset of the states provided to meet a specific set of applications. See “MPLF State transformation tables” on page 159 for a full set of states supported.

### Set Lock State processing

The Set Lock State order is used to set a Multi-Path Lock to a requested state (normally nonzero). On the basis of the current state of a given Multi-Path Lock, the requested state may or may not be granted. If the state is granted, the Perform Subsystem Function with this order is completed with presentation of channel end, device end, and status modifier ending status. If it is not granted, the Perform Subsystem Function with this order is completed with presentation of channel end and device end ending status. The result of the Set Lock State order can also be determined by use of the Read Subsystem Data command chained from the Set Lock State order.

The parameters that are used by the MPLF to determine if a requested lock state is granted or not are as follows:

#### Multi-Path Lock Filter Mask:

This parameter is specified in the Set Lock State order parameters. It defines the lock states that are compatible with the requested lock state.

#### Requested Lock State:

This parameter is specified in the Set Lock State order parameters. It defines the requested lock state.

#### Multi-Path Lock State:

This parameter is obtained from the Multi-Path Lock within the subsystem. It is the highest absolute numeric value lock state that is held by the current holder set for the given lock.

The State Transformation Tables under “MPLF State transformation tables” on page 159 can be used to determine if the granting of a lock is successful. To use the tables, find the table that pertains to the requested lock state. Determine the current Multi-Path Lock state by finding the highest absolute numeric value lock state that is held within the set of current holders of the lock, and use that value to select the table column. Determine the table row by using the Mask value that would be specified in the Set Lock State order. If the intersection of the row and column is an ‘xx’, the lock will not be granted. If the intersection is a numeric value, the lock is granted and the value is the resultant Multi-Path Lock state.

An example of this operation is given in Figure 47.

<p>Givens Case 1: LOCK STATE = 0101</p> <p>Lock States per Users</p> <table border="1"> <thead> <tr> <th>USER ID</th> <th>LOCK STATE</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>0001</td> </tr> <tr> <td>4</td> <td>0101</td> </tr> <tr> <td>7</td> <td>0010</td> </tr> </tbody> </table> <p>REQUEST: Set Lock issued by USER ID 6 Requested State = 0011, Mask = 0100</p> <p>PROCESSING: Lock State = max (0001, 0101, 0010) = 0101 Use table corresponding to a Requested State of 0011 Column: Lock State = 5 Row: Mask = 4 Intersection = xx</p> <p>RESULTANT STATES: LOCK STATE = 0101 REQUEST WAS NOT GRANTED</p>	USER ID	LOCK STATE	3	0001	4	0101	7	0010	<p>Givens Case 2: LOCK STATE = 0101</p> <p>Lock States per Users</p> <table border="1"> <thead> <tr> <th>USER ID</th> <th>LOCK STATE</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>0001</td> </tr> <tr> <td>4</td> <td>0101</td> </tr> <tr> <td>7</td> <td>0010</td> </tr> </tbody> </table> <p>REQUEST: Set Lock issued by USER ID 6 Requested State = 0011, Mask = 0111</p> <p>PROCESSING: Lock State = max (0001, 0101, 0010) = 0101 Use table corresponding to a Requested State of 0011 Column: Lock State = 5 Row: Mask = 7 Intersection = 5</p> <p>RESULTANT STATES: LOCK STATE = 0101 REQUEST WAS GRANTED</p>	USER ID	LOCK STATE	3	0001	4	0101	7	0010
USER ID	LOCK STATE																
3	0001																
4	0101																
7	0010																
USER ID	LOCK STATE																
3	0001																
4	0101																
7	0010																
<p>Givens Case 3: LOCK STATE = 0000</p> <p>REQUEST: Set Lock issued by USER ID (n) Requested State = ssss, Mask = mmmm</p> <p>PROCESSING: Lock State = 0 Use table corresponding to a Requested State of ssss Column: Lock State = 0 Row: Mask = mmmm Intersection = ssss</p> <p>RESULTANT STATES: LOCK STATE = ssss REQUEST WAS GRANTED</p> <p>LEGEND: n = any valid User ID (1-32) currently not a holder ssss = any valid Requested Lock State (0-15) mmmm = any valid Lock Mask (0-15) xxxx = any valid Lock State (0-15)</p>	<p>Lock States per Users</p> <table border="1"> <thead> <tr> <th>USER ID</th> <th>LOCK STATE</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>0001</td> </tr> <tr> <td>4</td> <td>0101</td> </tr> <tr> <td>6</td> <td>0011</td> </tr> <tr> <td>7</td> <td>0010</td> </tr> </tbody> </table> <p>Givens Case 4: LOCK STATE = xxxx</p> <p>REQUEST: Set Lock issued by USER ID (n) Requested State = ssss, Mask = 1111</p> <p>PROCESSING: Lock State = xxxx Use table corresponding to a Requested State of ssss Column: Lock State = xxxx Row: Mask = 15 Intersection = max (xxxx, ssss)</p> <p>RESULTANT STATES: LOCK STATE = max (xxxx, ssss) REQUEST WAS GRANTED</p>	USER ID	LOCK STATE	3	0001	4	0101	6	0011	7	0010						
USER ID	LOCK STATE																
3	0001																
4	0101																
6	0011																
7	0010																

Figure 47. Set Lock State example

**Promote/Demote Option:** The Promote/Demote option of the Set Lock State order allows a user to change the state of a lock for which the user is already a holder. Determine the current Multi-Path Lock state, as described in “Set Lock State processing” on page 156, but exclude the holder value for which this Promote/Demote option is being processed. This determined state is a temporary value. Use this temporary Multi-Path Lock state value in the tables, using the same procedures described in “Set Lock State processing” on page 156. If the Promote/Demote is not successful, the holder is restored to the value it had before execution of the Promote/Demote option, and the Multi-Path Lock state will retain its original value.

An example of this operation is given in Figure 48 on page 158.

<p>Givens Case 1: LOCK STATE = 0100</p> <p>Lock States per Users</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>USER ID</th> <th>LOCK STATE</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>0010</td> </tr> <tr> <td>3</td> <td>0011</td> </tr> <tr> <td>4</td> <td>0100</td> </tr> </tbody> </table> <p>REQUEST: Promote/Demote issued by USER ID 3 Requested State = 1000, Mask = 0001</p> <p>PROCESSING: Temporary Lock State = max (0010, 0100) = 0100 Use table corresponding to a Requested State of 1000 Column: Lock State = 4 Row: Mask = 1 Intersection = xx</p> <p>RESULTANT STATES: LOCK STATE = 0100 REQUEST WAS NOT GRANTED</p>	USER ID	LOCK STATE	2	0010	3	0011	4	0100	<p>Givens Case 2: LOCK STATE = 0100</p> <p>Lock States per Users</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>USER ID</th> <th>LOCK STATE</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>0010</td> </tr> <tr> <td>3</td> <td>0011</td> </tr> <tr> <td>4</td> <td>0100</td> </tr> </tbody> </table> <p>REQUEST: Promote/Demote issued by USER ID 4 Requested State = 1000, Mask = 0111</p> <p>PROCESSING: Temporary Lock State = max (0010, 0011) = 0011 Use table corresponding to a Requested State of 1000 Column: Lock State = 3 Row: Mask = 7 Intersection = 8</p> <p>RESULTANT STATES: LOCK STATE = 1000 REQUEST WAS GRANTED</p> <p>Lock States per Users</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>USER ID</th> <th>LOCK STATE</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>0010</td> </tr> <tr> <td>3</td> <td>0011</td> </tr> <tr> <td>4</td> <td>1000</td> </tr> </tbody> </table>	USER ID	LOCK STATE	2	0010	3	0011	4	0100	USER ID	LOCK STATE	2	0010	3	0011	4	1000
USER ID	LOCK STATE																								
2	0010																								
3	0011																								
4	0100																								
USER ID	LOCK STATE																								
2	0010																								
3	0011																								
4	0100																								
USER ID	LOCK STATE																								
2	0010																								
3	0011																								
4	1000																								

Figure 48. Promote/Demote example

### Unlock processing

The Unlock order results in the lock state for the specified user to be set to zero. The resultant state of the Multi-Path Lock is based on the state that the given Multi-Path Lock is held by other users.

An example of this operation is given in Figure 49 on page 159.



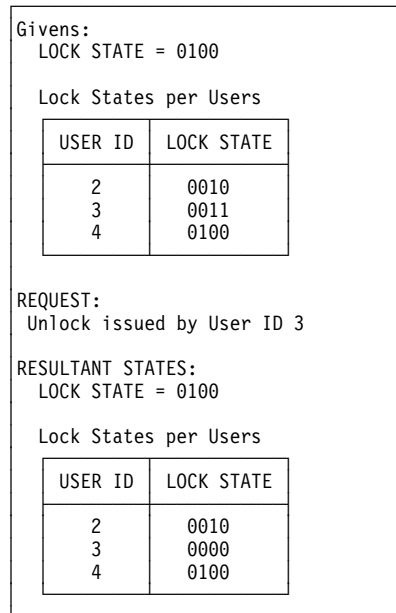


Figure 49. Unlock example

## Multi-Path Lock responses

The results of any Multi-Path Lock command can be detected inline in a channel program. If the requested order was successfully executed, Status Modifier is included in ending status, but if the requested order was not successfully executed, status modifier is not included in ending status. In addition to or instead of using the conditional setting of Status Modifier, the result of the operation can be retrieved in message form from the subsystem via a Read Subsystem Data command that follows the Multi-Path Lock request in the channel program.

The Read Subsystem Data command returns the data message associated with the last Perform Subsystem Function – Multi-Path Lock order executed. If a Read Subsystem Data command is not executed immediately following the Multi-Path Lock request, the message data is lost.

## Utility functions

The utility type functions provided follow:

- Perform Subsystem Function (X'27') – Order 31 Purge Lock. Purge a given Multi-Path Lock of a given holder, a given waiter, all holders, all waiters, or all waiters and holders.
- Perform Subsystem Function (X'27') – Order 22 Prepare to Read Lock Data. Retrieve information about the state of the partitions and locks within those partitions.

## MPLF State transformation tables

The following multi-part figure contains the Multi-Path Lock compatibility matrices. There is one table, within the figure, for each of the possible states that can be requested. Each table shows the effect with the 16 possible masks values. The first column, 'MASK', is the mask value that is used in conjunction with the

requested state. An 'xx' indicates that the requested state was not granted. A numeric value indicates the state was granted, and the value is the resultant state.

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
1	0	1	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
1	1	1	1	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
1	2	1	xx	2	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
1	3	1	1	2	3	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
1	4	1	xx	xx	xx	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
1	5	1	1	xx	xx	4	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
1	6	1	xx	2	xx	4	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx
1	7	1	1	2	3	4	5	6	7	xx	xx	xx	xx	xx	xx	xx	xx
1	8	1	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
1	9	1	1	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
1	10	1	xx	2	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
1	11	1	1	2	3	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
1	12	1	xx	xx	xx	4	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
1	13	1	1	xx	xx	4	5	xx	xx	8	9	xx	xx	12	13	xx	xx
1	14	1	xx	2	xx	4	xx	6	xx	8	xx	10	xx	12	xx	14	xx
1	15	1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
2	0	2	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
2	1	2	2	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
2	2	2	xx	2	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
2	3	2	2	2	3	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
2	4	2	xx	xx	xx	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
2	5	2	2	xx	xx	4	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
2	6	2	xx	2	xx	4	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx
2	7	2	2	2	3	4	5	6	7	xx	xx	xx	xx	xx	xx	xx	xx
2	8	2	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
2	9	2	2	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
2	10	2	xx	2	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
2	11	2	2	2	3	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
2	12	2	xx	xx	xx	4	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
2	13	2	2	xx	xx	4	5	xx	xx	8	xx	xx	xx	12	13	xx	xx
2	14	2	xx	2	xx	4	xx	6	xx	8	xx	10	xx	12	xx	14	xx
2	15	2	2	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Figure 50 (Part 1 of 6). Lock State Compatibility Matrix

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
3	0	3	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
3	1	3	3	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
3	2	3	xx	3	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
3	3	3	3	3	3	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
3	4	3	xx	xx	xx	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
3	5	3	3	xx	xx	4	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
3	6	3	xx	3	xx	4	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx
3	7	3	3	3	3	4	5	6	7	xx	xx	xx	xx	xx	xx	xx	xx
3	8	3	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
3	9	3	3	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
3	10	3	xx	3	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
3	11	3	3	3	3	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
3	12	3	xx	xx	xx	4	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
3	13	3	3	xx	xx	4	5	xx	xx	8	9	xx	xx	12	13	xx	xx
3	14	3	xx	3	xx	4	xx	6	xx	8	xx	10	xx	12	xx	14	xx
3	15	3	3	3	3	4	5	6	7	8	9	10	11	12	13	14	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
4	0	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
4	1	4	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
4	2	4	xx	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
4	3	4	4	4	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
4	4	4	xx	xx	xx	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
4	5	4	4	xx	xx	4	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
4	6	4	xx	4	xx	4	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx
4	7	4	4	4	4	4	5	6	7	xx	xx	xx	xx	xx	xx	xx	xx
4	8	4	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
4	9	4	4	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
4	10	4	xx	4	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
4	11	4	4	4	4	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
4	12	4	xx	xx	xx	4	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
4	13	4	4	xx	xx	4	5	xx	xx	8	9	xx	xx	12	13	xx	xx
4	14	4	xx	4	xx	4	xx	6	xx	8	xx	10	xx	12	xx	14	xx
4	15	4	4	4	4	4	5	6	7	8	9	10	11	12	13	14	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
5	0	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
5	1	5	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
5	2	5	xx	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
5	3	5	5	5	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
5	4	5	xx	xx	xx	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
5	5	5	5	xx	xx	5	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
5	6	5	xx	5	xx	5	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx
5	7	5	5	5	5	5	5	6	7	xx	xx	xx	xx	xx	xx	xx	xx
5	8	5	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
5	9	5	5	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
5	10	5	xx	5	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
5	11	5	5	5	5	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
5	12	5	xx	xx	xx	5	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
5	13	5	5	xx	xx	5	5	xx	xx	8	9	xx	xx	12	13	xx	xx
5	14	5	xx	5	xx	5	xx	6	xx	8	xx	10	xx	12	xx	14	xx
5	15	5	5	5	5	5	5	6	7	8	9	10	11	12	13	14	15

Figure 50 (Part 2 of 6). Lock State Compatibility Matrix

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
6	0	6	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
6	1	6	6	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
6	2	6	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
6	3	6	6	6	6	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
6	4	6	xx	xx	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
6	5	6	6	xx	xx	6	6	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
6	6	6	xx	6	xx	6	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx
6	7	6	6	6	6	6	6	6	7	xx	xx	xx	xx	xx	xx	xx	xx
6	8	6	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
6	9	6	6	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
6	10	6	xx	6	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
6	11	6	6	6	6	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
6	12	6	xx	xx	xx	6	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
6	13	6	6	xx	xx	6	6	xx	xx	8	9	xx	xx	12	13	xx	xx
6	14	6	xx	6	xx	6	xx	6	xx	8	xx	10	xx	12	xx	14	xx
6	15	6	6	6	6	6	6	6	7	8	9	10	11	12	13	14	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
7	0	7	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
7	1	7	7	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
7	2	7	xx	7	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
7	3	7	7	7	7	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
7	4	7	xx	xx	xx	7	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
7	5	7	7	xx	xx	7	7	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
7	6	7	xx	7	xx	7	xx	7	xx	xx	xx	xx	xx	xx	xx	xx	xx
7	7	7	7	7	7	7	7	7	7	xx	xx	xx	xx	xx	xx	xx	xx
7	8	7	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
7	9	7	7	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
7	10	7	xx	7	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
7	11	7	7	7	7	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
7	12	7	xx	xx	xx	7	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
7	13	7	7	xx	xx	7	7	xx	xx	8	9	xx	xx	12	13	xx	xx
7	14	7	xx	7	xx	7	xx	7	xx	8	xx	10	xx	12	xx	14	xx
7	15	7	7	7	7	7	7	7	7	8	9	10	11	12	13	14	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
8	0	8	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
8	1	8	8	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
8	2	8	xx	8	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
8	3	8	8	8	8	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
8	4	8	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
8	5	8	8	xx	xx	8	8	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
8	6	8	xx	8	xx	8	xx	8	xx	xx	xx	xx	xx	xx	xx	xx	xx
8	7	8	8	8	8	8	8	8	8	xx	xx	xx	xx	xx	xx	xx	xx
8	8	8	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
8	9	8	8	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
8	10	8	xx	8	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
8	11	8	8	8	8	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
8	12	8	xx	xx	xx	8	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
8	13	8	8	xx	xx	8	8	xx	xx	8	9	xx	xx	12	13	xx	xx
8	14	8	xx	8	xx	8	xx	8	xx	8	xx	10	xx	12	xx	14	xx
8	15	8	8	8	8	8	8	8	8	8	9	10	11	12	13	14	15

Figure 50 (Part 3 of 6). Lock State Compatibility Matrix

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
9	0	9	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
9	1	9	9	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
9	2	9	xx	9	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
9	3	9	9	9	9	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
9	4	9	xx	xx	xx	9	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
9	5	9	9	xx	xx	9	9	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
9	6	9	xx	9	xx	9	xx	9	xx	xx	xx	xx	xx	xx	xx	xx	xx
9	7	9	9	9	9	9	9	9	9	xx	xx	xx	xx	xx	xx	xx	xx
9	8	9	xx	xx	xx	xx	xx	xx	xx	9	xx	xx	xx	xx	xx	xx	xx
9	9	9	9	xx	xx	xx	xx	xx	xx	9	9	xx	xx	xx	xx	xx	xx
9	10	9	xx	9	xx	xx	xx	xx	xx	9	xx	10	xx	xx	xx	xx	xx
9	11	9	9	9	9	xx	xx	xx	xx	9	9	10	11	xx	xx	xx	xx
9	12	9	xx	xx	xx	9	xx	xx	xx	9	xx	xx	xx	12	xx	xx	xx
9	13	9	9	xx	xx	9	9	xx	xx	9	9	xx	xx	12	13	xx	xx
9	14	9	xx	9	xx	9	xx	9	xx	9	xx	10	xx	12	xx	14	xx
9	15	9	9	9	9	9	9	9	9	9	10	11	12	13	14	15	xx

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
10	0	10	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
10	1	10	10	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
10	2	10	xx	10	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
10	3	10	10	10	10	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
10	4	10	xx	xx	xx	10	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
10	5	10	10	xx	xx	10	10	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
10	6	10	xx	10	xx	10	xx	10	xx	xx	xx	xx	xx	xx	xx	xx	xx
10	7	10	10	10	10	10	10	10	10	xx	xx	xx	xx	xx	xx	xx	xx
10	8	10	xx	xx	xx	xx	xx	xx	xx	10	xx	xx	xx	xx	xx	xx	xx
10	9	10	10	xx	xx	xx	xx	xx	xx	10	10	xx	xx	xx	xx	xx	xx
10	10	10	xx	10	xx	xx	xx	xx	xx	10	xx	10	xx	xx	xx	xx	xx
10	11	10	10	10	10	10	xx	xx	xx	10	10	10	11	xx	xx	xx	xx
10	12	10	xx	xx	xx	10	xx	xx	xx	10	xx	xx	xx	12	xx	xx	xx
10	13	10	10	xx	xx	10	10	xx	xx	10	10	xx	xx	12	13	xx	xx
10	14	10	xx	10	xx	10	xx	10	xx	10	xx	10	xx	12	xx	14	xx
10	15	10	10	10	10	10	10	10	10	10	10	10	11	12	13	14	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
11	0	11	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
11	1	11	11	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
11	2	11	xx	11	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
11	3	11	11	11	11	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
11	4	11	xx	xx	xx	11	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
11	5	11	11	xx	xx	11	11	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
11	6	11	xx	11	xx	11	xx	11	xx	xx	xx	xx	xx	xx	xx	xx	xx
11	7	11	11	11	11	11	11	11	11	xx	xx	xx	xx	xx	xx	xx	xx
11	8	11	xx	xx	xx	xx	xx	xx	xx	11	xx	xx	xx	xx	xx	xx	xx
11	9	11	11	xx	xx	xx	xx	xx	xx	11	11	xx	xx	xx	xx	xx	xx
11	10	11	xx	11	xx	xx	xx	xx	xx	11	xx	11	xx	xx	xx	xx	xx
11	11	11	11	11	11	xx	xx	xx	xx	11	11	11	11	xx	xx	xx	xx
11	12	11	xx	xx	xx	11	xx	xx	xx	11	xx	xx	xx	12	xx	xx	xx
11	13	11	11	xx	xx	11	11	xx	xx	11	11	xx	xx	12	13	xx	xx
11	14	11	xx	11	xx	11	xx	11	xx	11	xx	11	xx	12	xx	14	xx
11	15	11	11	11	11	11	11	11	11	11	11	11	11	12	13	14	15

Figure 50 (Part 4 of 6). Lock State Compatibility Matrix

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
12	0	12	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
12	1	12	12	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
12	2	12	xx	12	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
12	3	12	12	12	12	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
12	4	12	xx	xx	xx	12	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
12	5	12	12	xx	xx	12	12	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
12	6	12	xx	12	xx	12	xx	12	xx	xx	xx	xx	xx	xx	xx	xx	xx
12	7	12	12	12	12	12	12	12	12	xx	xx	xx	xx	xx	xx	xx	xx
12	8	12	xx	xx	xx	xx	xx	xx	xx	12	xx	xx	xx	xx	xx	xx	xx
12	9	12	12	xx	xx	xx	xx	xx	xx	12	12	xx	xx	xx	xx	xx	xx
12	10	12	xx	12	xx	xx	xx	xx	xx	12	xx	12	xx	xx	xx	xx	xx
12	11	12	12	12	12	xx	xx	xx	xx	12	12	12	12	xx	xx	xx	xx
12	12	12	xx	xx	xx	12	xx	xx	xx	12	xx	xx	xx	12	xx	xx	xx
12	13	12	12	xx	xx	12	12	xx	xx	12	12	xx	xx	12	13	xx	xx
12	14	12	xx	12	xx	12	xx	12	xx	12	xx	12	xx	12	xx	14	xx
12	15	12	12	12	12	12	12	12	12	12	12	12	12	12	13	14	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
13	0	13	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
13	1	13	13	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
13	2	13	xx	13	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
13	3	13	13	13	13	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
13	4	13	xx	xx	xx	13	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
13	5	13	13	xx	xx	13	13	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
13	6	13	xx	13	xx	13	xx	13	xx	xx	xx	xx	xx	xx	xx	xx	xx
13	7	13	13	13	13	13	13	13	13	xx	xx	xx	xx	xx	xx	xx	xx
13	8	13	xx	xx	xx	xx	xx	xx	xx	13	xx	xx	xx	xx	xx	xx	xx
13	9	13	13	xx	xx	xx	xx	xx	xx	13	13	xx	xx	xx	xx	xx	xx
13	10	13	xx	13	xx	xx	xx	xx	xx	13	xx	13	xx	xx	xx	xx	xx
13	11	13	13	13	13	xx	xx	xx	xx	13	13	13	13	xx	xx	xx	xx
13	12	13	xx	xx	xx	13	xx	xx	xx	13	xx	xx	xx	13	xx	xx	xx
13	13	13	13	xx	xx	13	13	xx	xx	13	13	xx	xx	13	13	xx	xx
13	14	13	xx	13	xx	13	xx	13	xx	13	xx	13	xx	13	xx	14	xx
13	15	13	13	13	13	13	13	13	13	13	13	13	13	13	13	14	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
14	0	14	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
14	1	14	14	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
14	2	14	xx	14	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
14	3	14	14	14	14	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
14	4	14	xx	xx	xx	14	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
14	5	14	14	xx	xx	14	14	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
14	6	14	xx	14	xx	14	xx	14	xx	xx	xx	xx	xx	xx	xx	xx	xx
14	7	14	14	14	14	14	14	14	14	xx	xx	xx	xx	xx	xx	xx	xx
14	8	14	xx	xx	xx	xx	xx	xx	xx	14	xx	xx	xx	xx	xx	xx	xx
14	9	14	14	xx	xx	xx	xx	xx	xx	14	14	xx	xx	xx	xx	xx	xx
14	10	14	xx	14	xx	xx	xx	xx	xx	14	xx	14	xx	xx	xx	xx	xx
14	11	14	14	14	14	xx	xx	xx	xx	14	14	14	14	xx	xx	xx	xx
14	12	14	xx	xx	xx	14	xx	xx	xx	14	xx	xx	xx	14	xx	xx	xx
14	13	14	14	xx	xx	14	14	xx	xx	14	14	xx	xx	14	14	xx	xx
14	14	14	xx	14	xx	14	xx	14	xx	14	xx	14	xx	14	xx	14	xx
14	15	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	15

Figure 50 (Part 5 of 6). Lock State Compatibility Matrix

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
15	0	15	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
15	1	15	15	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
15	2	15	xx	15	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
15	3	15	15	15	15	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
15	4	15	xx	xx	xx	15	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
15	5	15	15	xx	xx	15	15	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
15	6	15	xx	15	xx	15	xx	15	xx	xx	xx	xx	xx	xx	xx	xx	xx
15	7	15	15	15	15	15	15	15	15	xx	xx	xx	xx	xx	xx	xx	xx
15	8	15	xx	xx	xx	xx	xx	xx	xx	15	xx	xx	xx	xx	xx	xx	xx
15	9	15	15	xx	xx	xx	xx	xx	xx	15	15	xx	xx	xx	xx	xx	xx
15	10	15	xx	15	xx	xx	xx	xx	xx	15	xx	15	xx	xx	xx	xx	xx
15	11	15	15	15	15	xx	xx	xx	xx	15	15	15	15	xx	xx	xx	xx
15	12	15	xx	xx	xx	15	xx	xx	xx	15	xx	xx	xx	15	xx	xx	xx
15	13	15	15	xx	xx	15	15	xx	xx	15	15	xx	xx	15	15	xx	xx
15	14	15	xx	15	xx	15	xx	15	xx	15	xx	15	xx	15	xx	15	xx
15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15

REQUESTED		CURRENT LOCK STATE															
LOCK	MASK	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
0	0	0	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
0	1	0	1	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
0	2	0	xx	2	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
0	3	0	1	2	3	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
0	4	0	xx	xx	xx	4	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
0	5	0	1	xx	xx	4	5	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
0	6	0	xx	2	xx	4	xx	6	xx	xx	xx	xx	xx	xx	xx	xx	xx
0	7	0	1	2	3	4	5	6	7	xx	xx	xx	xx	xx	xx	xx	xx
0	8	0	xx	xx	xx	xx	xx	xx	xx	8	xx	xx	xx	xx	xx	xx	xx
0	9	0	1	xx	xx	xx	xx	xx	xx	8	9	xx	xx	xx	xx	xx	xx
0	10	0	xx	2	xx	xx	xx	xx	xx	8	xx	10	xx	xx	xx	xx	xx
0	11	0	1	2	3	xx	xx	xx	xx	8	9	10	11	xx	xx	xx	xx
0	12	0	xx	xx	xx	4	xx	xx	xx	8	xx	xx	xx	12	xx	xx	xx
0	13	0	1	xx	xx	4	5	xx	xx	8	9	xx	xx	12	13	xx	xx
0	14	0	xx	2	xx	4	xx	6	xx	8	xx	10	xx	12	xx	14	xx
0	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Figure 50 (Part 6 of 6). Lock State Compatibility Matrix

**Note:** The lock state compatibility matrix in the bottom example (lock = 0) in Figure 50 (part 6 of 6) assumes that the current lock state is a transformation of the states for all other users of the lock. This matrix should not be confused with the Unlock order in terms of final status. A Set Lock State with a state of '0' and a mask of '15' is equivalent to an Unlock in terms of ending status.

## MPLF terminology

### Contention

If a given lock is held by user A, and another user, B, requests that lock, user B is causing contention. Under certain restrictive conditions, the user who is granted a state transformation other than zero, can request that the subsystem signals the user whenever any other user requests a state for the given Multi-Path Lock that cannot be granted. The signal is via an attention interrupt and is called a contention signal.

### Multi-Path Lock Name

An 8-byte user assigned name of a Multi-Path Lock. All character combinations except binary zeros are valid.

**Multi-Path Lock Filter Mask**

This mask identifies the lock states that are compatible with a requested Multi-Path Lock state.

**Demotion**

Demotion is the changing of a Multi-Path Lock state for a given user from a given state to a less restrictive state. The MPLF algorithms treat both Demotion and Promotion in the same manner because MPLF processing is not sensitive to state values.

**Free State**

See state 0.

**Holders**

The users of a given Multi-Path Lock who were granted the Multi-Path Lock in a state other than zero are the Holders of the Multi-Path Lock. The number of holders is limited to one per Multi-Path Lock User ID, that is a maximum of 32. The state in which the Multi-Path Lock is held is the state granted to the user that has the highest absolute numeric value.

**Multi-Path Lock Partition**

The Multi-Path Lock Partition is a means of logically subdividing the Multi-Path Locks for a user set. The Multi-Path Lock Partition is used as a high level qualifier for both the Multi-Path Lock User ID and the Multi-Path Lock Name. The maximum number of Multi-Path Lock Partitions is eight.

**Multi-Path Lock User ID**

The Multi-Path Lock User ID is an identifier of an MPLF user. The number of Multi-Path Lock User IDs is limited to 32 per Multi-Path Lock Partition. The Multi-Path Lock User ID is initially specified in the Connect order. The Connect order generates an expanded Multi-Path Lock User ID on the basis of the input value and returns the new version of the value in its completion message. The expanded Multi-Path Lock User ID is called the Multi-Path Lock User Token.

**Multi-Path Lock User Token**

The Multi-Path Lock User Token is an internally generated user identifier. It is generated by the Connect order and is used to identify users on MPLF orders that require authorization.

**Nonzero State**

If a Multi-Path Lock is in the nonzero state, the Multi-Path Lock is held by one or more users. If a Multi-Path Lock is held by multiple users, the state of the Multi-Path Lock is the state granted that has the highest absolute numeric value.

**Promotion**

Promotion is the changing of a Multi-Path Lock state for a given user from a given state to a more restrictive state. The MPLF algorithms treat both Demotion and Promotion in the same manner because MPLF processing is not sensitive to state values.

**Queue Option**

The user that requests a state transformation for a Multi-Path Lock that cannot be granted can optionally request that the subsystem queues that user as a waiter. When the requested Multi-Path Lock transformation is made, the user is signaled by presenting attention status on the device address established via the Connect order.



**State 0**

This is the free state of the Multi-Path Lock. If a Multi-Path Lock is in this state, any of the other states can be granted regardless of the Multi-Path Lock Filter Mask specified.

**Waiters**

If a user requests a state transformation for a Multi-Path Lock that cannot be granted and the user requested the queue option, that user is a waiter for the lock. The number of waiters is limited to 32. Usually, if there is more than one waiter for a Multi-Path Lock, they are serviced on a first-in first-out basis. Excluding Promotion and Demotion, if there is at least one waiter for a given Multi-Path Lock, no other user can be granted that Multi-Path Lock before that waiter is granted the Multi-Path Lock or was purged from the queue of waiters.



---

## Chapter 5. Sense conditions

ESS sense information consists of 32 bytes that completely identify the condition that caused the last Unit Check status presented for this device on this channel interface. The sense information also provides information that may be required for system error recovery. Other information is included to aid the customer engineer in diagnosing and isolating subsystem malfunctions and to notify the system operator of significant subsystem conditions.

The control unit sense is the 24 Byte Compatibility sense data. In both formats the additional sense bytes provide the device type identification and the pathing information needed for the new devices and control unit.

ESS continues SIM reporting for the Storage Control and all devices. All Storage Control and device Equipment Check and Data Check sense data will be logged by the Support Facility (SF). The SF data will include additional error dependent supporting data.

Thirty-two bytes of sense information are sent to the system on a Sense command. The bytes are organized by ESS when it detects an unusual condition that resulted in a Unit Check status. If the Sense command is not preceded by a Unit Check status for the same device on the same channel interface, the 32 bytes will be binary zeros except for the following:

<b>Byte</b>	<b>Contents</b>
<b>20-21</b>	Subsystem ID
<b>22-23</b>	Fault Symptom Code
<b>26-27</b>	Configuration Information

---

### Sense data summary

ESS sense data may be either 24 Byte Compatibility sense or ECKD 32 Byte sense format. The 24 Byte Compatibility mode format is indicated by byte 27, bit 0 being set to '1'.

24 Byte Compatibility sense data is used for the following:

- All control unit Equipment Checks
- All 'Program Checks' (Format 0) except:
  - End of extent with Imprecise Ending
  - Short Domain with Imprecise Ending

ECKD 32 Byte sense data is used for the following:

- All SIMs
- Device Equipment Check, Data Checks and Usage/Error Statistics.
- 'Program Exceptions' (Exception class 0) for:
  - End of extent with Imprecise Ending
  - Short Domain with Imprecise Ending

---

<sup>4</sup> Used by ESS in non-gap-synchronous mode

## 24 Byte Compatibility sense data summary

In 24 Byte Compatibility mode there are 4 sense formats in ESS. The first three bytes (bytes 0-2) and last eight bytes (bytes 24-31) of each format are identical. Sense bytes 3 through 6 are generally the same for all formats. Byte 7 of each format is defined as format/message. Bits 0-3 of this byte indicate the sense format used to assemble bytes 8-23.

General meaning for the various formats are given below and the details are in "24 Byte Compatibility sense data" on page 172.

Format	Category of Condition	Page
0	Programming or System Checks	180
2	ESS Equipment Checks	183
3	ESS Control Checks	184
F	Cache Storage Checks (ESS only.)	185

The sense bytes are defined as follows:

Figure 51. Sense information summary

Byte	Bit	Meaning	Page
0	0	Command Reject	172
	1	Intervention Required	
	2	Bus Out Parity Check	
	3	Equipment Check	
	4	Data Check	
	5	Overrun	
	6	Reserved	
	7	Incomplete Domain	
1	0	Permanent Error	172
	1	Invalid Track Format	
	2	End of Cylinder	
	3	Message to Operator	
	4	No Record Found	
	5	File Protected	
	6	Write Inhibited	
	7	Imprecise Ending	
2	0	Request Inhibit Write	172
	1	Correctable (data check)	
	2	First Error Log	
	3	Environmental Data Present	
	4	Reserved	
	5	Imprecise Ending	
	6	Reserved	
	7	Reserved	
3		Residual Count	176
4		Physical Drive ID	176
5		Low Cylinder Address	176
6		Head Address and High Cylinder Address	177
7	0-3	Format	177
	4-7	Message	
8-23		Format Dependent Information	180 - 185
24		Only used in ECKD 32-Byte format	178
25		Program Action Code	178
26-27		Configuration Information	178
28		Message Code	179
29-31		Cylinder and Head Address	179

## ECKD 32 Byte sense data summary

The contents of bytes 7-14 are dependent on the Exception Class in bits 0-3 of byte 22 and the format in bits 4-7 of byte 6. The other bytes contain the same information for all Exception Classes except as indicated in the following summary.

General meaning for the various formats are given below and the details are in "ECKD 32 Byte format" on page 189. The Exception Classes defined for ESS are:

Figure 52. ECKD 32 Byte Exception Class

Class	Definition	Page
0	I/O Program Exceptions	196
1	storage facility Diagnostic Detected (SIM only)	199
2-3	Storage Control Checks (SIM only)	199
4	Data Exceptions	199
5	Reserved	
6	Reserved	
7	RAS Microcode Checks (SIM only)	200
8-A	Reserved	
B	SCU/DA Interface Checks	200
C	DA Checks	200
D	DA/Device Interface Checks	200
E	Device Checks	200
F	Cache Storage Checks (SIM only)	201

Figure 53 (Page 1 of 2). ECKD 32 Byte sense information summary

Byte	Bit	Meaning	Page
0	0	Command Reject	189
	1	Intervention Required	
	2	Reserved	
	3	Equipment Check	
	4	Data Check	
	5	Reserved	
	6	Reserved	
	7	Incomplete Domain	
1	0	Permanent Error	190
	1	Invalid Track Format	
	2	Reserved	
	3	Message to Operator <sup>2</sup>	
	4	Reserved	
	5	File Protected	
	6	Write Inhibited	
	7	Imprecise Ending	
2	0-2	Control Unit Type or Must be zero <sup>2</sup>	191
	3	Environmental Data Present <sup>2</sup>	
	4-7	Must be zero <sup>2</sup>	
3		Remaining Intent Count or Command Overrun Count	191
4		Device Physical Address	191
5		Device Type Code	191
6		Content and Format	192
7-14		Exception Class and Format Dependent	195

Figure 53 (Page 2 of 2). ECKD 32 Byte sense information summary

Byte	Bit	Meaning	Page
15-19		Serial Number <sup>4</sup>	192
20-21		Subsystem ID	193
22-23		Symptom Code or	193
22	0-3	Exception Class	
	4-7	Exception Category	
23		Exception Code	
24		Flags	193
	0-2	Reserved	
	3	Logging Mode	
	4-5	Logging Control	
	6-7	Operator Message Control	
25		Program Action Code	194
26-27		Configuration-Dependent Information	194
28		Message Code <sup>3</sup>	195
29-30		Cylinder Address <sup>3</sup>	195
31		Head Address <sup>3</sup>	195

**Note:**

<sup>2</sup>Used in Exception class 1, 2, 3, 7, and F only.

<sup>3</sup>Bytes 28-31 are 'Bytes Read/Searched' for Exception class 6.

<sup>4</sup>Except Exception class 4 format 2.

## 24 Byte Compatibility sense data

### 24 Byte Compatibility Common sense bytes

Bytes 0-7 and Bytes 24-31 contain the same information for all '24 Byte Compatibility' sense formats.

#### Sense Bytes 0, 1, and 2

Sense Bytes 0–2 are **only** generated when a Unit Check is presented. These bytes describe the error condition in general terms and identify what specific action should be invoked to effect subsystem error recovery.

**Byte 0, Bit 0 Command Reject:** This is generated by any one of the following conditions (sense byte 7 identifies the error condition in more specific terms):

- An invalid command has been issued.
- A command has been issued to a primary or secondary address of a Duplex Pair that is invalid for the device status.
- An invalid sequence of commands has occurred.
- An invalid or incomplete argument has been transferred by a command.
- The Record 0 Count field of a track flagged as defective (or alternate) points to itself.
- A Write command has been issued that violates the File Mask.
- A command has been received in the domain of a Locate Record that does not conform with the operation parameter.

- The operation parameter of a Locate Record command specifies an operation inhibited by the file mask.
- A Format Write command other than Write HA, Write Special HA, or Write R0 was attempted on a defective track.
- A multitrack operation within the domain of a Locate Record, attempted to step to a cylinder that is outside the primary track group. File Protected (byte 1, bit 5) is also set.
- The status of the Cache Storage, the non-volatile storage or the device is not the status required for the Set Subsystem Mode or Perform Subsystem Function command.
- A Write Data operation with CKD conversion presented Channel End for the write command and found that the record length was incorrect. Invalid Track Format (byte 1, bit 1) is also set.

**Byte 0, Bit 1 Intervention Required:** This bit indicates that:

- The addressed device is in CE mode
- The addressed device is not ready
- The addressed device is an unconfigured LSA device
- The volume configuration of the addressed device is not correct

**Byte 0, Bit 2 Bus Out Parity Check:** This bit indicates that:

- An ESCON channel requested unit check with Channel Bus Out Parity Check sense or a 'selective reset' frame requesting retry and allowing unit check (with Channel Bus Out Parity Check sense) is received and retry cannot be attempted or is exhausted.

This error is presented as format 0, message A sense.

- An error was detected in a frame received from an ESCON channel and retry is exhausted or cannot be attempted.

This error is presented as format 2 sense.

- Channel Bus Out Parity Check termination was requested by the channel

This error is presented as format 0, message A sense.

**Byte 0, Bit 3 Equipment Check:** This bit indicates that an unusual hardware condition originated in the channel, storage facility or drive. The conditions that are covered by the bit are defined in Sense Bytes 1 through 23.

**Byte 0, Bit 4 Data Check:** This bit is not set in 24 Byte Compatibility sense.

**Byte 0, Bit 5 Overrun:** This bit indicates that one of the following occurred:

- An ESCON channel did not respond during a data transfer sequence or an invalid frame was received and the error is not retryable or retry is exhausted.

This error is presented as format 2 sense.

- A Channel Initiated Command Retry request allowing Overrun sense is received and the error is not retryable or retry is exhausted.

This error is presented as format 0, message A sense.

- Overrun termination was requested by an ESCON channel.

This error is presented as format 0, message A sense.

Detection of a data overrun will cause requests for data from the channel to be terminated. When writing, the remaining portion of the record area is padded with bytes of zeros. When possible, these conditions are retried by ESS. ESS posts data Overrun if the condition occurs more than ten times in a single CCW.

When an ESCON Channel Adaptor is dedicated to a recovery operation and the channel does not respond to Request In, the path is freed and overrun is reported to the next selection on the interface.

**Byte 0, Bit 7 Incomplete Domain:** Not set by ESS

**Byte 1, Bit 0 Permanent Error:** This bit is a modifier bit that overrides any other possible bit settings. If the sense data is associated with an active channel program, this bit indicates that redrive on the same channel path by system Error Recovery Procedures is not required. ESS sets this bit if:

Internal retry is not applicable and no other path is available within the Storage Director for system retry,

Internal error recovery has been exhausted and was unsuccessful for a Storage Director,

System error recovery is not possible or desirable.

If the operation is not associated with the current channel program and retry is exhausted for the Storage Control, Environmental Data Present (byte 2, bit 3) will be set with Permanent Error.

ESS does not set this bit on a data overrun that has been retried 10 times rather the system ERPs are called upon to retry the chain again.

**Byte 1, Bit 1 Invalid Track Format:** Invalid Track Format is set for any of the following reasons:

- An attempt was made to write data exceeding this tracks capacity.
- An update write operation is attempted on a record whose size differs from the record size parameter (TLF).
- An update write operation on R0 data field under Write Track intent is attempted on a field not equal to eight bytes in length.

Except for stage and destage errors, this sense is reported in Format 0, Message 0; some conditions are uniquely identified in byte 8 of that format.

When byte 1, bit 1 is set (Invalid Track Format), the residual Sub-Channel Status Word (SCSW) count provided in the interrupted CCW chain should not be used to determine the track capacity, because the residual count is not guaranteed on invalid track format.

There is no guarantee that any data transfer will take place in the key and data fields.

**Byte 1, Bit 2 End of Cylinder:** End of Cylinder indicates that a read or search multitrack operation has attempted to continue beyond the maximum head for a given cylinder. See Appendix A, "Device data, geometries, and facilities" on



page 219 for heads per cylinder for a given device. It should be noted that the maximum head is one less than the heads per cylinder number as the first head is head zero.

End of Cylinder usually indicates a programming error or an expected End of Cylinder condition has been detected.

**Byte 1, Bit 3 Message to Operator:** This bit means that a message should be sent to the operator.

**Byte 1, Bit 4 No Record Found:** No Record Found is set when the following conditions are detected:

- Two Index Points have been detected in the same CCW chain without an intervening read operation in the Home Address area or a Data area, or without an intervening Write, Sense, or Control command. When working with the device, ESS always verifies that the access mechanism is positioned properly before posting this condition. This bit usually indicates a programming error or an expected programming condition has occurred.
- The proper record can not be found as the result of the execution of a Locate Record Command. If the search for the record does not get executed until after Device End is posted for the Locate Record Command, the error will be presented during the following command execution and Imprecise Ending (byte 1, bit 7) will also be set.
- Within the domain of a Locate, after a head or cylinder switch either R0 or a User record could not be found.

**Byte 1, Bit 5 File Protected:** File Protected indicates one of the following has occurred:

- A seek, or multitrack operation outside the domain of a locate has violated the seek controls of the file mask.
- An attempt was made to access or operate on a track outside the boundaries established by a Define Extent command.

**Note:** If the condition is detected with Imprecise Ending, the sense format will be ECKD 32 Byte.

- A multitrack operation within the domain of a Locate Record, attempted to step to a cylinder that is outside the primary track group. Command Reject (byte 0, bit 0) is also set.
- A seek operation outside the primary track group.
- 'Set of tracks' outside the defined extent in Write Any or Read Any Extended Parameter
- Locate Record specifies a domain that would extend past extent boundaries
- Partition ID in an MPLF Set Lock State or Unlock is not the Partition authorized in the preceding Define Extent command.

File Protected indicates a programming error or an expected programming condition has been detected.

**Byte 1, Bit 6 Write Inhibited:** This bit is set whenever a valid write command is received on a logical path that was previously set in the inhibit write state by a Diagnostic Control command. Equipment Check (byte 0, bit 3) is also set.

**Byte 1, Bit 7 Imprecise Ending:** This bit is set to one when an abnormal channel program termination occurs within the domain of a Locate Record and the exception status is for a previously completed CCW (that is, one that has received ending status). This bit indicates that the SCSW command address is not in synchronism with the command actually being executed. This condition will exist when an error occurs while trying to do the search operation specified in a Locate Record Command if the search is not done before the next command is received or if the chain is operating in non-gap-synchronous mode. When byte 1, bit 7 is set (Imprecise Ending), the residual SCSW count provided in the interrupted CCW chain should not be used to determine the track capacity, because the residual count is not guaranteed.

This bit is duplicated in byte 2, bit 5, for compatibility with 3380 sense data.

**Byte 2, Bit 0, Request Inhibit Write:** This bit on indicates that the error recovery procedures should issue a command that will cause the subsystem to terminate any write commands with unit check and Write Inhibited sense if the error is not recovered on the path detecting the error. The area to be write inhibited is determined from the Program Action Code (Byte 25).

**Byte 2, Bit 1 Correctable:** Not set by ESS

**Byte 2, Bit 2 First Logged Error:** Not set by ESS

**Byte 2, Bit 3 Environmental Data Present:** This bit indicates that the sense information is not associated with the current CCW string, but rather is information that ESS needs to pass onto the operating system. Bytes 7 through 23 provide further information.

**Byte 2, Bit 5 Imprecise Ending:** This bit is a duplicate of byte 1, bit 7, and is retained for compatibility with current ERP.

### **Sense Byte 3, Residual Count**

This is the residual record/track count byte. With Imprecise Ending (byte 1, bit 7) set on, this byte indicates the number of records or tracks that remain to be processed within the domain of the Locate Record command.

For all other conditions, this byte is equal to zero.

### **Sense Byte 4, Physical Drive Identification**

This byte contains the physical device identification when a device is associated with the condition being reported. If a device is not associated with the sense data, this byte is zeros. This byte contains the unit address of the Base Volume.

### **Sense Byte 5, Low Cylinder Address**

**Note:** 3390-9 cylinder addresses cannot be contained in this byte. All references to 3390-9 cylinder / track addresses should be to sense bytes 29 to 31. For 3390-9, byte 5 will be X'FF'.

This byte identifies the low order cylinder address of the most recent seek argument accepted from the channel or an internal command chain for all formats except 6. If a device is not associated with the sense data, this byte is zeros.

The format of this byte follows:

- Bit 0 Cylinder address value 128
- Bit 1 Cylinder address value 64
- Bit 2 Cylinder address value 32
- Bit 3 Cylinder address value 16
- Bit 4 Cylinder address value 8
- Bit 5 Cylinder address value 4
- Bit 6 Cylinder address value 2
- Bit 7 Cylinder address value 1

### **Sense Byte 6, Head and High Cylinder Address**

**Note:** 3390-9 cylinder addresses cannot be contained in this byte. All references to 3390-9 cylinder / track addresses should be to sense bytes 29 to 31. For 3390-9, byte 6 will contain X'FF'.

This byte identifies the high order cylinder address and head address of the most recent seek argument accepted from the channel or an internal command chain for all formats except 6. If a device is not associated with the sense data this byte is zeros.

The format of this byte follows:

<b>Bit</b>	<b>Value</b>
<b>0</b>	Cylinder address value 2048
<b>1</b>	Cylinder address value 1024
<b>2</b>	Cylinder address value 512
<b>3</b>	Cylinder address value 256
<b>4</b>	Head address value 8
<b>5</b>	Head address value 4
<b>6</b>	Head address value 2
<b>7</b>	Head address value 1

### **Sense Byte 7, Format/Message**

Bits 0-3 of byte 7 identify the specific format of the remaining sense bytes as follows:

0-3 Format

- 0 Program or system checks
- 2 ESS equipment checks
- 3 ESS control checks
- F Cache Storage Miscellaneous Checks

For Formats 0, 2, and 3, bits 4-7 of byte 7 provide an encoded message that describes the specific nature of the error condition. The codes, which are unique for each sense format, are defined in the following section.

For Format F, bits 4 through 7 identify the message type and the specific format of bytes 8-23.

If Message to Operator (byte 1, bit 3) is on, this byte defines the message to be presented to the operator.

## Sense Bytes 8–24, Format Dependent Information

The description of these bytes relies on the sense data format. Sense data formats are described between 180 and 185.

### Sense Byte 24, Flags

This byte is only used in ECKD 32 Byte format. Values may be set but are not guaranteed in 24 Byte Compatibility sense mode.

### Sense Byte 25, Program Action Code

This byte contains information that indicates to the Error Recovery Programs (ERPs) what type of action to take. The actions are described in Figure 58 on page 204 and in the accompanying text.

In 24 Byte Compatibility format this byte is only used by the ERP in only the following specific cases. Other codes may be set but will not be used.

Bit 0 is '0' and bits 1-7 define specific program actions.

**Bits 1-7** These bits define specific program actions for unique conditions.

#### Hex

#### Value Set when:

- 00** No unique condition.
- 16** Reset Notification, Equipment Check and byte 7 = X'08'
- 18 or 58** Equipment Malfunction during write operation. If permanent on this path, write operations should be inhibited for the Channel Path.
- 1D** A state change pending condition exists for the subsystem or device. Environmental Data Present (byte 2, bit 3) is set and byte 7 = F0.
- 70** A Global Command was rejected because the interface was disabled for Global Commands by a Perform Subsystem Function, Set Special Intercept Condition order. The interface on which the command was rejected is no longer disabled. A Retry will not be rejected for this reason.
- 71** An attention was generated and a Unit Check was added to the status because the interface was disabled for Global Commands by a Perform Subsystem Function, Set Special Intercept Condition order. The interface on which the attention and unit check was presented is still disabled.

### Sense Bytes 26-27, Configuration-Dependent Data

This bit significant field contains information to identify unique conditions of the subsystem. The definition of the 'on' condition for each bit when bit 0 of byte 27 is set to '1' (24-byte compatibility sense format) is:

#### Bit Description

- 26-0** Zero
- 26-1** One (DLSE for 3990-03)
- 26-2,3** Zero
- 26-4** Zero
- 26-5** Zero
- 26-6** Put in System Exception Report. This bit is not used in 24 Byte Compatibility sense
- 26-7** Zero
- 27-0** 24-byte compatibility sense data format.

- 27-1 Device address in byte 4 is valid
- 27-2 Track address valid in bytes 29-31
- 27-3 Reserved
- 27-4 Device is in Track Emulation Mode or transition to Track Emulation Mode .
- 27-5 Reserved
- 27-6 Cluster 1
- 27-7 Zero

### Sense Byte 28, Message Code

This byte, in conjunction with the format, defines the operator message. If bit 0 of byte 28 is 1, the message is a SIM message that is defined in “Sense Byte 28, Message Code” on page 195 (ECKD 32 Byte).

The message codes for non-SIM data (24 Byte Compatibility) are set based on the condition detected by the Storage Control. The messages are defined in combination with the format (byte 7, bits 0-3), The non-SIM message codes (MSG CODE) are defined as follows:

Figure 54. Coded message code generation

Byte 7	MTO	Description	MSG CODE
F9	on	Caching Reinitialized	09
FB	on	Volume VVVVV is Failed Duplex	0B
FC	on	Volume VVVVV Status Cannot be Determined	0C
FD	on	Caching Status Reset to Default	0D
FE	on	Defective Battery	0E

#### Notes:

1. See “Messages Defined by Sense” on page 210 for details on message contents.

### Bytes 29-31, Cylinder and Head Address

These bytes identify the cylinder and head address of the most recent seek argument accepted from the channel or executed by an internal operation. The format of these bytes is as follows:

Byte-Bit	Value
29-0-1	Reserved
-2	Cylinder 8192
-3	Cylinder 4096
-4	Cylinder 2048
-5	Cylinder 1024
-6	Cylinder 512
-7	Cylinder 256
30-0	Cylinder 128
-1	Cylinder 64
-2	Cylinder 32
-3	Cylinder 16
-4	Cylinder 8
-5	Cylinder 4
-6	Cylinder 2
-7	Cylinder 1

<b>31-0-3</b>	Reserved
<b>-4</b>	Head 8
<b>-5</b>	Head 4
<b>-6</b>	Head 2
<b>-7</b>	Head 1

## Format 0 - Program or System Check

With Message to Operator (byte 1, bit 3) set off, this format reports program or system check conditions (in byte 7). With Message to Operator set on, this format reports the end of a soft-error logging sequence or the application of an internal path fence.

### Format 0 messages

#### *Without MTO:*

1. Message 0 - No Message

This message is generated if the program check is defined in bytes 0-2. Additional information (RC) may be provided in byte 8.

2. Message 1 - Invalid Command

This message is generated when a command is issued that is not in ESS's command set. See Figure 5 on page 19 for the valid commands.

3. Message 2 - Invalid Command Sequence

This message is generated when an invalid sequence of commands has occurred.

See specific commands for error messages.

4. Message 3 - CCW Count less than Required

This message is generated when the CCW count of a command is less than required. The following are included:

- a. Any Seek command with a count less than 6.
- b. A Set Path Group Identifier command with a count less than 12.
- c. A Write Home Address command with a count less than 5.
- d. A Write Special Home Address command with a count less than 28 bytes.
- e. A Locate Record command with a count less than required.
- f. A Define Extent command with a count less than required.
- g. A Perform Subsystem Function command with a count less than the number of bytes required.
- h. A Set Subsystem Mode command with a count less than 2 bytes.
- i. A Diagnostic Control command with a count less than 4 bytes.

See specific commands for error messages.

5. Message 4 - Invalid Parameter

This message is generated when a data argument of a command is invalid. See specific commands for error messages.

6. Message 6 - Reserved

7. Message 7 - Channel Returned with Incorrect Retry CCW

This message is generated when the command portion of the CCW returned after a command retry sequence does not match the command for which retry was signaled.

#### 8. Message 8 - Reset Notification

This message is generated whenever a system reset or its equivalent is received on an interface. The Unit Check that generates this sense is posted to the next channel initiated selection following the resetting event.

**Note:** For the first selection to the interface, Environmental Data Present and Equipment Check are set. The first selection of other devices on that interface receives this format 0, message 8 sense with Equipment Check only. The Program Action Code (Byte 25) contains X'16'.

#### 9. Message 9 - Storage Control Restart

An operation for an active channel program was queued in a Storage Control when a warm start was received by the path. Equipment Check (byte 0, bit 3) is set.

#### 10. Message A -

There are several reasons for this message; byte 8 contains the reason codes as follows:

##### **Code Reason**

- 00** No Message
- 01** The channel has requested unit check sense data.
- 02** The channel has requested retry and retry is exhausted.
- 03** A SA Check-2 error has occurred. This sense is presented with Equipment Check.
- 04** The channel has requested retry and retry is not possible.

#### 11. Message B - Invalid Defective/Alternate Track Pointer.

This message is generated if a defective track is accessed.

**Note:** The subsystem generates an invalid Defective/Alternate Track Pointer as a part of RAID Recovery.

The sense is presented with Command Reject.

#### 12. Message D - Reserved

#### 13. Message F - Status Not As Required.

There are several reasons for this message; byte 8 contains the reason code defined as follows:

##### **Code Reason**

- 00** No Message
- 04** Attempt to activate Cache Fast Write Access and Cache Fast Write Access status is pending.
- 06** Attempt to Destage Modified Tracks with the non-volatile storage status 'failed' or attempt to 'Discard' DASD Fast Write data with 'data in the failed NVS for the device'.
- 09** Attempt to Force Deactivate DASD Fast Write for Device and there is no pinned data for the volume in the non-volatile storage.

- 1D** A Diagnostic Control command with an order other than X'0B' (Diagnostic Initialize Subsystem) was received on a subsystem that was in 'FC' condition (Status Cannot be Determined) or a Diagnostic Control with an order other than X'0F' was received on a device that had an unreadable Partition Status Area.
- 20** A Diagnostic Control command with a 'conditional' Diagnostic Initialize Subsystem order was received and the subsystem has a valid Global subsystem status track.
- 22** A device specified in byte 2 or 3 of an is in CE Mode or 'Status Cannot be Determined' state (Sense format F, Message C).
- 25** An attempt to 'Make Cache Storage Unavailable for Subsystem' or 'Make non-volatile storage Unavailable for Subsystem' while that storage is initializing.
- 28** Attempt to Read Pinned Tracks and the Cache Storage is failed or unavailable.
- 29** Attempt to execute a Perform Subsystem Function, Set Special Intercept Condition order on an interface on which no path group has been established.
- 2C** A Diagnostic Control, Diagnostic Initialize Subsystem is received and a previous Diagnostic Initialize Subsystem is in progress.
- 2E** A PSF Commit or other destage request cannot be performed because Internal Command Chains are not allowed as a result of a permanent device error.
- 2F** A command was chained from a Reset Allegiance that was unsuccessful.
- 30** A command chained from a Sense that bypassed device busy encounters the device busy condition.
- 34** Interface ID not a channel interface for this subsystem.
- 3B** The mode of the device does not match the mode required by the subcommand modifier of Diagnostic Control command with the Emulation Control subcommand.
- 49** A command other than Sense ID, Read Device Characteristics, Read Configuration Data, Sense Subsystem Status, Read Subsystem Data, or Perform Subsystem Function (with an order of prepare to read subsystem data) was received on a device that was busy and the I/O was started with either a Sense ID, Read Device Characteristics, Read Configuration Data, or a Sense Subsystem Status command.
- 4C** The MPLF partition is invalid or not initialized.
- 4D** The MPLF user token is not valid or not defined for this channel interface and Partition.
- 80** Attempt to execute a Global Command on an interface that was disabled for Global Commands by a Perform Subsystem Function, Set Special Intercept Condition order.
- 81** An attention was presented on an interface that was disabled for Global Commands by a Perform Subsystem Function, Set Special Intercept Condition order.



- 91 A Perform Subsystem Function Prepare to Read Lock Data order was received requesting Multi-Path Lock Facility data and the data was not available for presentation because: either the requested Multi-Path Lock Partition is not initialized, or cache is not available for use.
- 93 A cache storage control command cannot be accepted now because a Multi-Path Lock Disconnect operation is in progress for an initialized Multi-Path Lock Partition.
- 95 A Multi-Path Lock Facility order that can generate an Attention message cannot be accepted because the subsystem message buffer is full.

**With MTO:** The following codes request operator messages. The message code in Message Code (Byte 28) is the same as byte 7.

**Message Meaning**

- 0 No Message
- 1-3 Reserved
- 4 Data Pinned for the device

Modified data in cache or the non-volatile storage exists for the device and cannot be destaged to the device. This is the first track pinned for this device.

**Note:** For this message, the CCH values in bytes 5-6 and 29-31 are indefinite and do not identify the pinned track.

- 5-F Reserved

**Bytes 8-23 for Format 0**

The following is a description of the Format 0 sense bytes.

**Bytes Definition**

- 8 Contains 'Reason Code' in some messages, otherwise zero. See "Byte 1, Bit 1 Invalid Track Format" on page 174 and Message F in the preceding section.
- 9-14 Zeros
- 15-19 Serial Number  
Sense bytes 15 through 19 contain the plant of manufacture and serial number information. This data is obtained from the VPD.
- 20-21 Subsystem ID.
- 22-23 Fault Symptom Code (FSC)

**Format 2 - ESS Equipment Check**

This sense format is generated when a ESS failure occurs that is not associated with the Cache Storage or non-volatile storage and does not inhibit the Storage Path from reporting it to the system.

**Messages for Format 2**

When bits 0-3 of sense byte 7 specify format 2, bits 4-7 are defined as:

**Code Definition**

- 0-7 Reserved
- 8 Storage Control Check 2
- 9-D Reserved

- E RAS Microcode errors detected by Storage Control
- F Microcode Detected Errors, Error Code in byte 8

### **Sense Bytes 8-23 for Format 2**

The following is a description of the Format 2 sense bytes.

#### **Bytes Definition**

**8** Reason Code for Message F, otherwise zero

**9-14** Zeros

**15-19** Serial Number

Sense bytes 15 through 19 contain the plant of manufacture and serial number information. This data is obtained from the VPD.

**20-21** SSID

The SSID of the affected Logical Subsystem obtained from the VPD.

**22-23** Fault Symptom Code (FSC)

## **Format 3 - ESS Control Check**

This sense format is generated when a channel operation has been terminated by a Reset Allegiance or Unconditional Reserve command.

### **Messages for Format 3**

The following messages are defined for sense format 3:

**0-E** Reserved

**F** Connection broken as the result of an Reset Allegiance or Unconditional Reserve command on another channel. This error is reported with Equipment Check.

**Note:** The Reset Allegiance and Unconditional Reserve are normally issued by the Missing Interrupt Handler (MIH).

### **Sense Bytes 8-23 for Format 3**

The following is a description of the Format 3 sense bytes.

#### **Bytes Definition**

**8-14** Zeros

**15-19** Serial Number

Sense bytes 15 through 19 contain the plant of manufacture and serial number information. This data is obtained from the VPD.

**20-21** SSID

The SSID of the effected Storage Control obtained from the VPD.

**22-23** Fault Symptom Code (FSC)

## Format F - Subsystem Data

This format is used when errors or conditions associated with various functions are detected.

The message in bits 4-7 of byte 7 determine the error type or condition, as follows:

- Message 0, Operation Terminated  
The subsystem terminates an operation related to an active channel program when termination and redrive are required and logging is not desired.
- Message 1, Subsystem Processing Error  
A microcode logic error is detected
- Message 2, Cache Storage or non-volatile storage Equipment Failure  
An equipment failure is detected in the Cache Storage or non-volatile storage.
- Message 4, Reserved
- Message 6, Cache Fast Write Access Not Authorized  
A request for Cache Fast Write Data access does not match the access currently authorized for the subsystem.
- Message 7 Reserved
- Message 8, Reserved
- Message 9, Caching Reinitiated  
Caching has been automatically reinitiated following an error.
- Message C, Volume Status Cannot be Determined  
The status of a DASD Fast Write or Duplex Pair volume cannot be determined.
- Message D, Caching Status Reset to Default
- Message E, Defective Battery  
DASD Fast Write is not allowed because of an non-volatile storage battery check condition.
- Message F, Reserved

### Sense Bytes 8-23 for Format F

For some format F messages, the contents of byte 8 is assigned as 'Reason Code'. For the remainder of the messages, the contents of 8-21 are the same as format 2 unless other information is specified for a particular message (or message and reason code). In all cases bytes 22-23 contain the Fault Symptom Code. The reason codes are 'Message Code' dependent. See "Sense Bytes 8-23 for Format 2" on page 184 for the definition of bytes 9-23.

### Format F, Message 0 'Operation Terminated'

This message is generated when the subsystem rejects or terminates an operation related to an active channel program.

This sense data is not generated for any internal operation that is terminated but that is not associated with an active channel program. This sense record is used only to cause the attaching host to reissue the request and, in the case of State Change Pending, to request a queueing of the request and a redrive of the request on a periodic basis. Environmental Data Present (byte 2, bit 3) is set.

**Note:** Environmental Data Present is set only to insure redrive of the operation. The sense record does not contain environmental data and should not be logged.

**Reason Codes for Format F, Message 0 - '1D':** Reason codes are specified when this message is the result of a State Change Pending condition and the Program Action Code (Byte 25) is set to '1D'. When this message is generated for reasons other than a State Change Pending condition, byte 8 is set to '00'.

**Code Reason (byte 8)**

- 00** Not a State Change Pending condition.
- 01** Emergency destage in progress.
- 02** Diagnostic Reinitialization.
- 03-06** Reserved
- 07** A PSF-Commit order was received.
- 08-0B** Reserved
- 0C-0F** Reserved

**Format F, Message 1 'Subsystem Processing Error'**

This message is generated when a microcode logic error that is associated with the Cache Storage, the non-volatile storage, or the caching function is detected.

If the error results in caching termination or reinitialization and is associated with the current channel program, Equipment Check (byte 0, bit 3) is set. If the error does not cause caching termination or reinitialization, but is associated with the current channel program, Equipment Check only is set in bytes 0-2.

If the error was detected by an operation other than the current operation, Equipment Check and Environmental Data Present (byte 2, bit 3) are set.

Byte 8 contains Reason Codes and bytes 9-10 contain additional data assigned by microcode.

**Format F, Message 2 'Cache Storage Equipment Failure'**

This message is generated when an unrecoverable error is detected in the Cache Storage or non-volatile storage operation.

If the error is not recovered by internal error handling procedures within the Storage Control it is reported with Equipment Check (byte 0, bit 3) set.

If system retry is not desirable, Permanent Error (byte 1, bit 0) and Equipment Check are set on and Environmental Data Present (byte 2, bit 3) set off.

If the error was detected by an operation other than the current operation, Permanent Error (byte 1, bit 0) and Environmental Data Present (byte 2, bit 3) are also set.

**Reason Codes for Format F, Message 2:** Byte 8 contains the Error Type in bits 0-3 and an Operation Type in bits 4-7.

The Error Types are:

- Bits**
- 0-3 Error Type**
  - 0** Reserved

- 0-E Reserved
- F Microcode Detected Error

The Operation Types are defined as follows:

**Bits**

**4-7 Operation Type**

- 0 No Operation Type
- 1-F Reserved

**Format F, Message 6 ‘Cache Fast Write Access Not Authorized’**

This sense is generated when:

- A Define Extent command with the Cache Fast Write Data attribute active or a Perform Subsystem Function with a Commit order is received and the Cache Fast Write Access ID in the command parameters is not the current CFWID for the subsystem. In this case, Command Reject (byte 0, bit 0) is set.
- A Cache Fast Write track that is pinned in cache because the cache image is unusable is accessed. In this case, Equipment Check (byte 0, bit 3) and Permanent Error (byte 1, bit 0) is set.

**Reason Codes for Format F, Message 6:**

**Code Description**

- 0 Reserved
- 01 CFWID is not the current ID.
- 02 Track is Pinned and unusable

**Note:** For this error the track address should be included in sense bytes 29-31.

03-FF Reserved

**Format F, Message 9 ‘Caching Reinitiated’**

This message is generated when the Storage Director automatically reestablishes the caching function following an error or condition. If the reinitialize is the result of an error, another Equipment Check sense message defining the failure that caused caching reinitialization accompanies this sense message. The message is only sent if caching had been active prior to the failure.

Message to Operator and Environmental Data Present only are set in bytes 0-2.

If approximately 16,000 Start I/Os have not been received on an interface following an IML of the Storage Director, or a System Reset on that interface when the sense data is generated, the sense data presentation is delayed until they have been received. This delay is to insure that system message handling capability is available when the sense data is presented.

**Reason Codes for Format F, Message 9:**

- | Code | Meaning  |
|------|--|
| 00   | No message                                       |
| 01   | Diagnostic Initialize Subsystem received         |
| 02   | Subsystem failure                                |
| 0A   | Storage Control to Cache Storage Equipment Check |
| 0B   | FC state entered with known status               |
| 0E   | Internal microcode restart of the storage path   |

0F-FF Reserved

### **Format F, Message C ‘Status Cannot be Determined’**

This message is generated when the Storage Controller cannot locate the Global subsystem status tracks, cannot establish the status of a DASD Fast Write volume, cannot access the RAID rank that the data is stored on, or cannot synchronize the volume with data in the non-volatile storage.

Depending on the condition of the error, the sense is presented to all but selected commands on all affected devices until the condition is cleared. The conditions that may produce this sense, and the commands that are accepted for each condition are:

- If the Global subsystem status cannot be determined at IML or cannot be updated while operational, the Subsystem is ‘FC’ and all addresses receive unit check.
- If the condition keeps a RAID rank from being operational then all commands other than Locate Record, Locate Record Extended, Seek, Seek Cylinder, and Read IPL commands are accepted by the affected device.

The initial presentation of this sense for each device includes Equipment Check (byte 0, bit 3), Permanent Error (byte 1, bit 0), and Message to Operator (byte 1, bit 3) in bytes 0-2 of the sense data. All subsequent sense has only Equipment Check and Permanent Error set in bytes 0-2. The Message Code (Byte 28) is ‘0C’.

#### ***Reason Codes for Format F, Message C:***

Code Meaning

##### **Device FC State**

- 00 Not Determined, if the condition persists subsequent sense contains a valid code.
- 01 RAID Rank is not available
- 02-04 Reserved
- 05 Partition Status Area Data Check or format error
- 06 Modified data may exist in NVS for device and the data is not available in Cache Storage is not available in Cache Storage
- 07 RAID Rank is not available and modified data may exist in NVS for the device.
- 08 NVS has been repaired with data loss and modified data may exist in NVS for the device.
- 09-0F Reserved

##### **Subsystem Known FC State**

- 10 Reserved
- 11 Reconfigure CNT = 0 or a diagnostic subsystem INIT command is being processed, and a Subsystem Status Partition could not be established.
- 12 Error writing Subsystem Status Partition and no other valid RAID ranks are available to replace the Subsystem Status Partition.
- 13-7F Reserved

##### **Subsystem Unknown FC State**

- 80-BF Reserved
- C0-EF Reserved
- F0-FF Valid, bits 4-7 decoded as:
  - 00xx** Device 1 Not Ready
  - 01xx** Device 1 Status Track Unreadable

**10xx** Device 1 Identifier Invalid  
**11xx** Device 1 Global Status Valid  
**xx00** Device 2 Not Ready  
**xx01** Device 2 Status Track Unreadable  
**xx10** Device 2 Identifier Invalid  
**xx11** Device 2 Global Status Valid

### **Format F, Message E ‘Defective Battery’**

This message is generated when DASD Fast Write is inhibited while the NVS is available or pending. and when the non-volatile storage is made available while a defective battery condition exists.

**Note:** A SIM sense message is also sent when the NVS battery condition is initially detected by the RAS Microcode.

Equipment Check (byte 0, bit 3), Environmental Data Present (byte 2, bit 3) and Message to Operator (byte 1, bit 3) are set in bytes 0-2.

If approximately 16,000 Start I/Os have not been received on an interface following an IML of the Storage Director, or a System Reset on that interface when the sense data is generated, the sense data presentation is delayed until they have been received. This delay is to insure that system message handling capability is available when the sense data is presented.

**Reason Codes for Format F, Message E:** There are no reason codes for this error.

---

## **ECKD 32 Byte format**

The ECKD 32 Byte format is used in ESS for SIM sense messages and sense data for 3390 and RAMAC devices. The definition of sense data when 24 Byte Compatibility is off is as follows:

### **ECKD 32 Byte Common sense bytes**

Sense bytes 0-6 and 15-31 are the same for all ECKD 32 Byte sense Exception classes.

#### **Sense Byte 0**

Bits 0–7 descriptions follow:

**Byte 0, Bit 0, Command Reject:** This bit is set with Incomplete Domain (byte 0, bit 7) when the expected number of data transfer commands was not received within the domain of a Locate Record or Locate Record Extended, if no other error was detected.

**Byte 0, Bit 1, Intervention Required:** This bit indicates that the addressed device is in CE mode.

**Byte 0, Bit 3, Equipment Check:** This bit indicates that an unusual hardware condition originated in the DASD Controller or drive.

**Byte 0, Bit 4, Data Check:** This bit indicates that a data error has been detected in the information received from the drive. This bit is set if the data error is uncorrectable or if correction is inhibited by the mask byte.

**Byte 0, Bit 7, Incomplete Domain:** When '1', this bit indicates that the expected number of data transfer commands, as specified in the Locate Record count parameter, were not received by the subsystem. This indicator may be set in addition to other indicators in bytes 0 and 1.

**Note:** When an Incomplete Domain exception occurs, it may prevent the unit check from a subsequent exception from being presented with ending status for the operation.

This indicator is set to indicate that an Incomplete Domain did occur and may be included in sense data representing some other exception condition.

**Note:** If Permanent Error (sense byte 1, bit 0) is set with Incomplete Domain, then write the Permanent Error message to the operator.

## Sense Byte 1

Bits 0–7 descriptions follow:

**Byte 1, Bit 0, Permanent Error:** When '1', this bit indicates an exception condition has been detected in a component of the reporting unit. Any internal retry count or recovery action available to correct this component failure has been exhausted and the exception condition persists.

**Byte 1, Bit 1, Invalid Track Format:** Invalid Track Format is set for any of the following reasons:

- An update write operation is attempted on a record whose size differs from the record size parameter (TLF).

**Byte 1, Bit 3, Message to Operator (MTO):** This bit is only set for control unit SIMS (Exception class 1, 2, 3, 7, and F) except for 'Trace Saved', if the severity in byte 9, bits 0-1 is equal to or greater than the SIM Severity Option parameter specified in Vital Product Data. When Message to Operator bit is set, byte 24, bits 6-7 = '01' (message unconditional) is also set. It indicates that a SIM message should be sent to the operator.

**Byte 1, Bit 5, File Protected:** When '1', this bit indicates that an attempt was made to step to a track that is not within the defined extent during a Locate Record Write operation. Imprecise Ending (byte 1, bit 7) may also set. Byte 22, bits 0-3 indicate an I/O Program exception and bits 4-7 indicate a Protection exception. See "Exception class 0, I/O Program Exception" on page 196.

**Byte 1, Bit 6, Write Inhibited:** This bit is set under the following conditions:

- A valid write command is received by ESS that must use a resource that was previously set in the inhibit write state by a Diagnostic Control command. Equipment Check (byte 0, bit 3) is also set.
- A RAID rank is Write Inhibited (Exception class E Exception Category F)

**Byte 1, Bit 7, Imprecise Ending:** This bit is set to '1' when an abnormal channel program termination occurs within the domain of a Locate Record and the exception status is for a previously completed CCW (that is, one that has received ending status). This bit indicates that the SCSW command address is not in synchronism with the command actually being executed by ESS. This condition occurs in ESS when an error occurs while trying to do the search operation specified in a Locate Record command, if the search is not done until ESS has



received the next command or if the chain is operating in non-gap-synchronous mode.

In most cases, this bit is set along with other bits in bytes 0-1 to indicate that the error is not associated with the CCW that returned the unit check. It is set without any other error when:

- An update write within the domain of a Locate Record steps from a track that is a cache 'miss' to a cache 'hit' and the track is modified or 'busy' or
- A channel program ends and the count for the domain has not been satisfied.

### **Sense Byte 2, Control Unit Type or Environmental Data Present**

For control unit SIMs (Exception class 1, 2, 3, 7, and F) this byte is reserved, except for bit 3. Environmental Data Present (byte 2, bit 3) is set in SIM data for ERP compatibility.

For 3390 Exception class 4, B, C, D, and E errors (including SIMs), this byte contains an eight-bit code that identifies the control unit type, model and features. This is the same code that appears in byte 42 of the data returned by the Read Device Characteristics command. See Appendix A, "Device data, geometries, and facilities" on page 219.

### **Sense Byte 3, Count**

When the exception condition reported by this sense data occurred within the domain of a Locate, this byte contains the number of the 'count' parameter from the preceding Locate Record command, minus the number of transfer units (records or tracks) that were successfully transferred to the device or to the channel within an interrupted Locate Record domain.

If this is SIM-format (byte 6, bits 4-7=X'F') sense data, byte 3 is defined as:

*Figure 55. Definition of Byte 3 for SIM-Format Sense Data*

Bits	Value	Definition
0-3	0	Old SIM (used by ESS)
	1	DASD SIM
	2	DASD EIM (Environmental Information Message)
	3-F	Not Used
4-7		Reserved

Otherwise, this byte is '00'.

### **Sense Byte 4, Device Physical Address**

Unit address of the Volume

### **Sense Byte 5, Device Type Code**

For Exception class 4, B, C, D, and E, this byte has a code that defines the device type and model. This code is the real device code, not the emulated one. The value of the code is obtained from byte 41 of the Read Device Characteristics command if byte 57 is zero. If byte 57 of the Read Device Characteristics command is non-zero, this code is the same as in byte 57. If the device type is not known, the code for the base device type is set to (X'26'). See Appendix A, "Device data, geometries, and facilities" on page 219.

This byte is zero for Exception class 1, 2, 3, 7, and F.

### Sense Byte 6, Content and Format

This byte indicates the presence of certain optional fields and defines the format of bytes 7-19 as follows:

Figure 56. Sense Byte 6, Content and Format

Bit	Value	Description
0		Serial number valid. This bit is always zero for Exception Classes 0 and 6 it is always one for Exception class 1, 2, 3, 7, and F.
	0	Serial number is not valid.  In addition, for Exception class 4, B, C, D, and E, the reporting unit is the unit identified in byte 2.
	1	Serial number in bytes 15-19 is valid.  In addition, for Exception class 4, B, C, D, and E, the reporting unit is the unit identified in byte 5.
1		Device address valid.
		When one, sense byte 4 is valid.
2		Track address valid
		When one, the Track Address in bytes 29-31 is valid
3		Must be zero
4-7		Format of bytes 7-19 for the Exception Class. A format of 'F' defines a SIM message for all Exception Classes.

### Sense Bytes 7-19, Exception Class-Dependent Data

Bytes 7-14 are SIM-format sense data that is described by Exception Class under "Exception Class dependent data" on page 195; bytes 15-19 are described under "Sense Bytes 15-19, Serial Number."

### Sense Bytes 15-19, Serial Number

For Exception Class 0, the content of these bytes is format dependent. See "Exception class 0, I/O Program Exception" on page 196 for their definition.

For Exception class 1, 2, 3, 7 and F, this field contains the storage facility Serial Number.

For Exception class 4, B, C, D, and E bytes 15-19 contain the serial number of the reporting unit. If the reporting unit is the device then the serial number reported will be that of the DDM drawer.

If the serial number is unknown, these bytes are zeros.

**Definition of Serial Number:** Serial numbers are obtained from 5 bytes and shown as mmaa-nnnnn where:

mm: is the manufacturer

Byte 15, bits 0-5 are a binary number that identifies the manufacturer; '000001' = IBM

aa: is the plant of manufacture (byte 15, bit 6 through byte 16, bit 7)

nnnnn: is the manufacturing sequence number (bytes 17-19)

## Sense Bytes 20-21, Subsystem ID

Bytes 20 and 21 contain the SSID of the affected Logical Subsystem. The SSID is obtained from the VPD.

## Sense Bytes 22-23, Exception Code

The Exception code contains a detailed description of the condition that caused the sense to be assembled. Byte 22, bits 0-3 contain the 'Exception Class' code. The remainder of the field depends on the Exception Class and further defines the error. The Exception Classes defined for ESS are summarized under "ECKD 32 Byte sense data summary" on page 171.

The definition of the Exception class and format dependent data in bytes (7-14) are detailed in "Exception Class dependent data" on page 195.

## Sense Byte 24, Flags

These flags are defined as follows:

Bit	Definition
0-2	Reserved
3	Logging Mode
4-5	Logging Control
6-7	Operator Message Control

**Logging Control:** These bits instruct the ERP program on how to log this error. These bits are defined as follows:

00	Do not log
01	Log (Unconditional)
10	Log only once
11	Log only if persistent

The logging action is to be performed in addition to any other program action that is specified. These bits are the only means to request sense data logging. A program must not be required to interpret additional sense data to determine if logging is required, but may interpret other bits to determine the format of the data to be logged.

When byte 24, bits 4-5 = '01', the sense data is always logged.

When byte 24, bits 4-5 = '10', the sense data is logged once and only once within a retry sequence on a path, whether the retry action is successful or not.

When byte 24, bits 4-5 = '11', the sense data is logged only if the error condition persists through all retry actions on a path.

For SIM sense data, these bits are '01', 'log unconditionally'.

**Operator Message Control:** Byte 24, bits 6-7 indicate the conditions under which a message is to be sent to the operator. Byte 28 contains a message code that, in combination with the Exception Class in byte 22, bits 0-3, uniquely identifies the message to be sent.

The conditions follow:

00	Send <i>no</i> message to the operator
01	Send a message to the operator (Unconditional)

- 1 0** Send message once
- 1 1** Send message only if error persists

The message action is to be performed in addition to any other program action that is specified. These bits are the *only* means to request a specific message action. A program must not be required to interpret additional sense data to determine if an operator message is required.

When byte 24, bits 6-7 = '10', a message is to be sent to the operator once, and only once within a retry sequence on a path, whether the retry action is successful or not.

When byte 24, bits 6-7 = '11', a message is to be sent to the operator only if the error condition persists through all retry actions on a path. The following conditions set non-zero values:

1. For SIM sense data, other than 'Trace Saved', these bits are '01', 'send console message unconditionally' if the severity (byte 9, bits 0-1) is equal to or greater than the SIM Severity Option parameter specified in Vital Product Data; otherwise, it is '00'.
2. Data Check on the primary device of a Duplex Pair recovered on the secondary sets '01'.
3. Intervention Required, device switch disabled or motor power off sets '01'.
4. A storage controller is 'fenced' sets '01'.
5. A device is 'fenced' and the RAS Microcode is not available sets '01'.

### **Sense Byte 25, Program Action Code**

This byte contains information that indicates to the Error Recovery Programs (ERPs) what type of action to take. See "Coded program actions" on page 213 for details on the codes and the recover actions.

### **Sense Byte 26-27, Configuration Data**

This bit significant field contains information to identify unique conditions of the subsystem. The bit set indicates:

<b>Bit</b>	<b>Description</b>
<b>26-0</b>	Zero
<b>26-4</b>	Zero
<b>26-5</b>	Zero
<b>26-6</b>	Put in System Exception Report. This bit is not used in 24 Byte Compatibility sense
<b>26-7</b>	Zero
<b>27-0</b>	24-byte compatibility sense data format (off in this format).
<b>27-1</b>	Device address in byte 4 is valid
<b>27-2</b>	Track address valid in bytes 29-31
<b>27-3</b>	Reserved
<b>27-4</b>	Device is in Track Emulation Mode or transition to Track Emulation Mode.
<b>27-5</b>	zero
<b>27-6</b>	Cluster 1
<b>27-7</b>	Zero

## Sense Byte 28, Message Code

Except for Exception class '6' and format 'F' (SIM), this byte is defined as the 'Message Code' or 'Additional Error Action' code. This byte defines the message if the Operator Message Control (byte 24, bits 6-7) is not zero and defines additional ERP action if the Program Action Code has bits 0 (Compound PAC) and bit 2 (Additional error action) set.

Non-SIM messages in ECKD 32 Byte sense are listed in Figure 57.

Figure 57. Non-SIM message codes

EXCP Class	CONDITION	Byte 28	MAC
02	Locate Record Incomplete Domain	01	01
4x	Data check recovered on secondary device of a Duplex Pair	01	01
EF	Intervention Required	01	01

The ERP action is defined in "Compound Program Action Codes" on page 213.

## Sense Bytes 29-30, Cylinder Address

For Exception classes 4 and B through E (except format 'F'), these bytes contain the cylinder address of the most recent seek argument accepted from the channel or executed by an internal operation. Byte 6, bit 2 is set to indicate that this field is valid.

See "Format F, SIM Data" for other contents of these bytes.

## Sense Byte 31, Head Address

For Exception classes 4 and B through E (except format 'F'), these bytes contain the head address of the most recent seek argument accepted from the channel or executed by an internal operation. Byte 6, bit 2 is set to indicate that this field is valid.

See "Format F, SIM Data" for contents of this byte for SIMs.

## Exception Class dependent data

The specific error type for ECKD 32 Byte sense is identified by the Exception class in byte 22, bits 0-3. This class determines the meaning of the 'format' in bits 4-7 of byte 6 and, with the format, the contents of the format dependent bytes.

### Format F, SIM Data

For all Exception Classes, if the format in byte 6 is 'F' the contents of bytes 7-14 is SIM sense data. For Exception class 1, 2, 3, 7, and F, Environmental Data Present (byte 2, bit 3) are always set and Message to Operator (byte 1, bit 3) may be set. Permanent Error (byte 1, bit 0) is set if the error is permanent. For all SIM data, the contents of bytes 7-14 are as follows:

**Byte 7, SIM Sense Record ID** This is reported in binary coded decimal (values of 00 - 99).

**Byte 8, Exception/Service Message Code:** Bits 0-3 of this byte contains a code indicating the severity of the failure and bits 4-7 contains a code indicating the impact of repair. Codes are dependent on the Message Code in byte 28.

This is set to '8F'x for all Exception classes except for Exception class 4. This is a unique code that asks for the CE to be called, it does not indentify any failure or repair impacts.

For Exception class 4 this is set to '12'x.

**Bytes 9 and 10, Message Modifiers:** This byte contains flags that provide additional information on the failure and repair impact.

Byte/Bit	Description
9 0-1	SIM Severity Code
9 2-7	Exception/Service message modifiers
10 0	Repeat SIM indicator
10 1	SIM presented against unrelated resource
10 2-7	Reserved

**Byte 28, SIM Code:** The SIM code defines the message to be sent when the Operator Message Control (byte 24, bits 6-7) is non-zero. The following messages are defined. This code is set whether or not a console message is requested in the Operator Message Control (byte 24, bits 6-7).

**Value Message Type**

F1	Storage Control Unit (SCU) Service SIM
F2	Cache Storage Service SIM
F3	SCU Remote Session SIM
FE	Device Service SIM
FF	Device Customer SIM

**Bytes 29-31, SIM Message Modifiers:** Set to zero

**Exception class 0, I/O Program Exception**

This class of exception reports errors or conditions during non-gap-synchronous operations that require unique Error Recovery Procedures. This Exception class has four Exception Categories:

Bits 4-7	Category
0010	Command Sequence Exception
0101	Protection Exception

**Exception Category 2, Command Sequence Exception:** This Exception is reported when the expected number of data transfer commands is not received in the domain of a Locate Record or Locate Record Extended command and no other exception was detected in completing the domain activity.

This error is reported as follows:

- Format (byte 6, bits 4-7) = X'4'
- Bytes 0-1 = X'0100' (Incomplete Domain)
- Program Action Code (Byte 25) = X'10'
- Exception Code (byte 23) = X'12'.

**Exception Category 5, Protection Exception:** This Exception is reported when a Locate Record domain extends beyond the extent boundary.

This error is reported as follows:

- Format (byte 6, bits 4-7) = X'2'

- Bytes 0-1 = X'0005' (File Protected and Imprecise Ending)
- Program Action Code (Byte 25) = X'15'
- Exception Code (byte 23) = X'02'

**Format 2 for Exception class 0:** Format 2 presents sense data associated with an I/O program exception that occurred within the domain of a Locate Record command, the Program Action Code specifies retry of the operation and the Search Argument specifies the last record successfully transferred.

Bytes 3, 7-9, and 11-18 contain Locate Record parameter values that may be used in constructing a Locate Record parameter to restart the operation.

**Bytes Contents**

<b>7</b>	Operation byte
<b>8</b>	Auxiliary byte
<b>9</b>	Locate Record parameter byte 2
<b>10</b>	Reserved
<b>11-15</b>	Search argument
<b>16</b>	Sector number
<b>17-18</b>	TLF
<b>19</b>	Extended Operation Byte

**Bytes 7-9** are the values received in the Locate Record parameter.

**Bytes 11-15** contain the Record ID of the last record successfully transferred or, if no records were transferred, the search argument from the Locate Record parameter.

**Byte 16** contains the sector number value to be used in the restart Locate Record.

**Bytes 17-18** contain the Transfer Length Factor.

**Byte 19** When the Operation byte (byte 7) contains X'3F', this byte contains the extended operation code from byte 17 of the original command, otherwise this byte is unassigned.

**Note:** When an I/O exception occurs within the domain of a Locate Record Extended command, the error recovery program must determine from the contents of byte 19, whether an Extended Parameter was present in the original command or not. If the recovery action involves the length or content of an Extended Parameter, the program must obtain this data from the original command within processor storage, since this information is not available in the sense data.

Error recovery programs may be unable to find the original Locate Record Extended command when the command follows a part of a channel program that contains conditional branching (via Status Modifier). In this case, no recovery action may be possible.

**Bytes 29-31** contain the track address 'CCH' at which the exception occurred, or if the exception was detected before track accessing was initiated, the seek address from the Locate Record parameter. If the exception occurred on an alternate track, bytes 29-31 contain the address of the assigned defective track. Byte 6, bit 2 is '1' to indicate the presence of the track address.

**Format 4 for Exception class 0:** Format 4 presents sense data associated with an exception caused by a Locate Record domain containing fewer data transfer commands than were required to transfer the number of records or tracks specified by the count field of the Locate Record parameter (Incomplete Domain). This exception is presented in response to a subsequent initial selection to the same device when the exception is not recognized until after ending status had been presented.

This format is presented with Command Reject (byte 0, bit 0) and Incomplete Domain (byte 0, bit 7) in sense bytes 0-1 and the “count” in sense byte 3 is the remaining domain count.

**Note:** The data presented in this format should provide information relevant to identifying the channel program that caused the exception.

The contents of the format-specific bytes are:

Byte	Bit	Contents
6	4-7	X'4' Format
7-8		Locate Record Operation code
9-12		Beginning Extent address
13-19		Path Group ID
22	0-3	X'0' Exception class
	4-7	X'2' Exception category
23		X'12' Symptom code
24	4-5	binary '01' Log unconditionally
	6-7	binary '01' Send message unconditionally
25		X'10' Program Action Code (PAC)
28		X'01' Message code

**Byte 0, Bit 7** is set to '1' to indicate that an Incomplete Domain exception occurred.

**Byte 3** contains the number of records or tracks remaining to be transferred when the Incomplete Domain exception occurred.

**Byte 7** contains the operation code from parameter byte 0 of the Locate Record or Locate Record Extended command that established the domain.

**Byte 8** contains the extended operation code from parameter byte 17 when the domain was established by a Locate Record Extended command; otherwise, it contains binary zero.

**Bytes 9-12** contain the Beginning of Extent address from the Define Extent command in the channel program.

**Bytes 13-19** contain bytes 1-7 of the Path Group ID parameter transmitted by the most recent Set Path Group Identifier command to the channel path. If no Path Group ID exists, these bytes contain binary zeros.

**Bytes 29-31** contain the track address 'CCH', at which the exception occurred. If the exception occurred on an alternate track, bytes 29-31 contain the address of the assigned defective track.



### **Exception class 1, Diagnostic Detected**

This format is used when errors are detected by diagnostic checkout. The Exception class is only reported in the SIM format defined in "Format F, SIM Data" on page 195.

### **Exception class 2, Storage Control Checks**

This format reports that errors or conditions associated with the Storage Controls are detected. The Exception class is only reported in the SIM format defined in "Format F, SIM Data" on page 195.

### **Exception class 3, Storage Control Checks**

This format reports that errors or conditions associated with the Storage Controls are detected. The Exception class is primarily reported in the SIM format defined in "Format F, SIM Data" on page 195.

### **Exception class 4, Data Checks**

This class of error is DASD Data Checks.

This Exception class is generated under the following conditions:

1. ECC data checks in an area that has had data sent to the channel and that has been successfully corrected during retry but the system File Mask indicates PCI fetch. In this PAC will be set to X'1C'
2. SIM messages for media maintenance are reported as format 'F'.
3. Data was damaged during a write operation or reconstruct operation. Data Check (byte 0, bit 4) is set. In this PAC will be set to X'80'

Exception class 4 is reported using format 1 (byte 6, bits 4-7=1).

#### **Bytes Contents**

<b>7</b>	Sector Number. This field is unreliable if the count field is unreadable.
<b>8-12</b>	The record ID from the count field in which the error occurred. This field is unreliable if the count field is unreadable.
<b>13-14</b>	Displacement to first byte in error. For Exception Category 8, these bytes are zero.
<b>15-19</b>	Serial number of the unit on which the exception occurred.

**Bytes 22-23 for Exception class 4:** The Exception Categories in bits 4-7 of byte 22 for Exception class 4 are:

#### **4-7 Definition**

<b>0</b>	Data Check in Home Address
<b>1</b>	Data Check in Count Field
<b>2</b>	Data Check in Key Field
<b>3</b>	Data Check in Data Field
<b>8</b>	Invalid data
<b>9</b>	Data Exception Service Message

The Exception Conditions in byte 23 for Exception Categories 0-7 are:

#### **Bit Description**

<b>0-1</b>	ECC Correction Bits
<b>00</b>	Recovered without using ECC

- 01 Recovered using First Level Correction
- 10 Recovered using Second Level Correction
- 11 Permanent Error, Not Recovered
- 2 Offset Active
- 3 Reserved (set = binary '0')
- 4 Offset Capability Exceeded in One Direction
- 5 DA Voltage Condition (DA Power Status bit 4)
- 6 DA Thermal Condition (DA Power Status bit 5)
- 7 Loss of Fine Track (NOT Servo Status 0 Low bit 5)

The Exception Conditions in byte 23 for Exception Category 8 are:

Value	Description
00	No Exception Conditions
01	Indeterminate event
02	System reset
03	Selective reset
04	Interface disconnect or cancel
05	Attachment failure
06	System interface transmission failure
07	Data was lost during a media repair action.
08	RAID Adapter Failure
09-FF	Reserved

### Exception class 7, RAS Microcode Checks

This format is used when errors or conditions associated with the RAS Microcode and the IML device are detected by the RAS Microcode. The Exception class is only reported in the SIM format defined in "Format F, SIM Data" on page 195.

### Exception class B, SCU/DA Interface

This class of error is storage facility/Device Adaptor Interface errors detected by the DA or Storage Control. The error is always presented with Equipment Check (byte 0, bit 3) set for non-SIM conditions.

### Exception class C, DA

This class of error is Controller Check 2 errors detected by the DA. The error is always presented with Equipment Check (byte 0, bit 3) set for non-SIM conditions.

### Exception class D, Controller/Device Interface

This class of error is Controller/Device Interface errors detected by the DA. The error is always presented with Equipment Check (byte 0, bit 3) set for non-SIM conditions.

### Exception class E, Device

This class of error is device errors. The error is always presented with either Equipment Check (byte 0, bit 3) or Intervention Required (byte 0, bit 1) set for non-SIM conditions.

### Exception Categorys 0-E

#### Exception Category F

**Volume, Adapter, DDM, RAID Rank state conditions:** Bytes 7-14 will be zero.

Byte 23 will have the following definition

LAC will be set to B'01'.

**Value Definition**

- 0** No exception condition
- 1** RAID Rank is write inhibited. Most probable reason is that a DDM is not available and sparing cannot be invoked.  
PAC will be set to X'80'.
- 2** RAID Rank has been taken offline by CE action (Intervention Required (byte 0, bit 1) is set for this condition).  
PAC will be set to X'02'.
- 4** RAID Rank is write inhibited, another RAID Rank in this subsystem has data trapped in a failed NVS.  
PAC will be set to X'80'.
- 6** The configuration information for this volume is not correct. Intervention Required (byte 0, bit 1) is set for this condition.  
PAC will be set to X'80'.
- 80-FF** Reserved for internal microcode conditions. These reason codes are not expected to be presented in sense data to a host.

**Exception class F, Cache Storage Checks**

This format reports errors or conditions associated with the Cache storage or Nonvolatile Storage. The Exception class is only reported in the SIM format defined in "Format F, SIM Data" on page 195.



---

## Chapter 6. Error recovery procedure

ESS reports errors in two sense formats. The sense format determines the Error Recovery Action.

Devices use the Coded ERP actions described in “Coded program actions” on page 213 for sense data in the ECKD 32 Byte sense format. This is sense from devices and SIM sense messages.

For other than SIMs, the control unit reports sense data in the 24 Byte Compatibility format and uses the information in bytes 0–2 to determine recovery requirements.

---

### Bytes 0-3, ERP description

The Error Condition Table presented below identifies all unique configurations of sense bits (in bytes 0-2) that are posted by the ESS and maps each of these configurations into a specific recovery action to be invoked by the system. The table provides a column of cross-references to “Sense Bytes 0, 1, and 2” on page 172 to assist in identifying the specific error or unusual conditions that generated the sense bit configuration.

The Recovery Action Table provided later in this section describes the error recovery action referenced in the Error Condition Table. These recommended actions assume that the Sense information was collected by a Sense command without command chaining (see “Sense” on page 86).

### Error condition table

If a recording action is specified, the system must create one of the following records:

- A permanent Data Check recovered on the secondary of a Duplex Pair is reported as Environmental Data with Message to Operator.
- MDR type 90 (Miscellaneous Data Record) for records read from Read and Reset Buffered Log Command.
- MDR type 91 for records presented as format 6 from a unit check
- A3 for SIM records.
- OBR type 30 for all other records.

Priority of testing the following conditions is at the discretion of the Error Recovery Program (ERP).

Error conditions causing unique configurations of sense bits and multiple error conditions are not posted concurrently. All valid combinations in bytes 0-2 are shown except that Request Inhibit Write (byte 2, bit 0), Imprecise Ending (byte 1, bit 7) and Message to Operator (byte 1, bit 3) are not included and may be set with several listed bit combinations.

Request Inhibit Write does not affect the initial ERP action exit but may be examined within the recovery action to determine subsequent steps.

When Imprecise Ending is set, the error action for File Protected (action 10) must use the count field from byte 3 of sense to determine the failing CCW when rebuilding a Locate Record domain.

Message to Operator requests a console message defined by the format and message in byte 7.

Figure 58 (Page 1 of 2). Error Condition Table

Byte Bit	Sense Data	Condition	Fmt	Act	Log
0 0	Command Reject	Programming error or invalid defective track pointer detected	0, F6	2	NO
0 0 1 5	Command Reject File Protected	Attempt to step to a cylinder outside the primary functional track group or used an MPLF Partition not authorized by the preceding Define Extent command.	0	2	NO
0 0 2 3	Command Reject Environmental Data Present	Attempt to execute a Global CCW on an interface disabled by a Perform Subsystem Function, Set Special Intercept Condition order. (PAC 70)	0F	10	NO
0 1	Intervention Required	The device is not ready.	0	3	NO
0 2	Channel Bus Out Parity Check	Bus out parity error or BOPC requested by the channel.	0, 2	3	YES <sup>1</sup>
0 3	Equipment Check	Equipment malfunction or Subsystem processing error.	0, 2, F1, F2	5	YES <sup>1</sup>
0 3 1 0	Equipment Check Permanent Error	Equipment Malfunction - retry exhausted or undesirable (Condition related to current channel program)	F2	1	YES
0 3 1 0 2 3	Equipment Check Permanent Error Environmental Data Present	Equipment Malfunction, Communication failure, microcode logic error or controller fenced. Retry exhausted.	2, F2, FB	4	YES <sup>1</sup>
0 3 2 3	Equipment Check Environmental Data Present	Equipment Check recovered by internal ERP, Caching or NVS Terminated. Controller fenced with additional paths available, trace table saved or DASD Fast Write inhibited.	0, 2, F1, F2, F4, FA, FE	4	YES <sup>1</sup>
0 3 1 6	Equipment Check Write Inhibited	Write Operations Inhibited on this data path by a Diagnostic Control command	0	9	NO
0 5	Overrun	Command retry exhausted on a service overrun or overrun error requested by channel.	0, 2	5	YES <sup>1</sup>
1 1	Invalid Track Format	Track capacity exceeded (Condition related to current channel program)	0, F7	2	NO
1 1 2 3 1 0	Invalid Track Format Environmental Data Present Permanent Error	Track Format error on destage, not associated with an active channel program	F7	4	NO
1 2	End of Cylinder	End of cylinder detected during a M/T operation	0	7	NO
2 3	Environmental Data Present (except byte 7 = '0F')	Pinned Data, subsystem redrive required, long busy condition, or a Subsystem Storage or NVS message or a SIM.	0, 2, F	4	YES <sup>1</sup>
2 3	Environmental Data Present (byte 7 = '0F')	Attention was presented on an interface disabled by a Perform Subsystem Function, Set Special Intercept Condition order. (PAC 71)	0F	11	NO
1 4	No Record Found	Record not found during command sequence (Condition related to current channel program)	0	2	NO
1 5	File Protected	Seek, M/T operation or Perform Subsystem Function order has violated the file mask or extent limits	0	8	NO

Figure 58 (Page 2 of 2). Error Condition Table

Byte Bit	Sense Data	Condition	Fmt	Act	Log
1 0	Permanent Error	Service Information Message (SIM)	E0	4	YES
2 3	Environmental Data Present				

**Note:**

<sup>1</sup> Not logged if any of the following:

- The format (bits 0-3 of byte 7) is X'0', except when the message (bits 4-7 of byte 7) is X'4' and Message to Operator (byte 1, bit 3) is on; or
- The format is X'F' and Message to Operator (byte 1, bit 3) is on; or
- The format is X'F' and the message is X'0.'

## Channel error conditions

All interface control checks (including Disconnect In) and channel control checks cause Action 4 for all termination and retry codes.

## Error recovery actions

The Recovery Action Table specifies an Error Correction Function (ECF) as a necessary step in the recovery procedure. The Error Correction Function section provides the ECF algorithm and related instructions that are required.

Where a TIC (CCW) is specified in the Recovery Action Table, the ERP must test the CCW to which the SCSW is pointing. If the SCSW is pointing to a user's TIC, the ERP must TIC to the command address specified in the user's TIC. This is done to prevent a TIC to a TIC that is a program check.

For all actions: if Message to Operator is set, a console message defined in "Messages Defined by Sense" on page 210 should be printed for operator and/or CE notification.

### Action 1

If all alternate paths have not been tried, repeat the operation using a different path.

If all alternate paths have been tried, post the job with a permanent error (refer to "Message 1" on page 209).

### Action 2

Exit with programming error or unusual condition indication.

### Action 3

1. Repeat the operation once.

If the error is Intervention Required, issue a message to the operator indicating that intervention is required on the reporting device, then queue the I/O request until an interrupt is received from the device indicating the device has been made ready.

2. If the error condition persists, then do Action 1.

## Action 4

1. If the Program Action Code (Byte 25) is '1D', the control unit or device is in a 'state change pending' condition.

Redrive the request when the subsystem signals the end of a state change pending via a State Change Interrupt.

**Note:** If the host is timing I/O operations, the request may be re-driven at some period greater than 15 seconds. The subsystem continues to return 'state change pending' sense as long as the operation is in progress.

2. Repeat the operation up to 256 times.
3. If the error condition persists, then do Action 1.

**Note:** The only reason to retry the operation more than once is to allow for queued environmental sense data within the subsystem. This allows the subsystem to unit check channel-initiated selections to get messages to the operator or pertinent sense data logged in the system.

## Action 5

1. If ERP does not support Reset Notification, skip to step 3.
2. If sense byte 7 = '08', the Path-Group-ID for this path has been reset by some other means other than a SPID command. The system must ensure that this device is still grouped as expected for this system before the current task is reissued.
3. If the chain has not been retried 10 times, repeat the user's command chain.
4. If the error condition persists after ten retries, determine if Request Inhibit Write (byte 2, bit 0) was off for all executions of the user's chain. If it was off for all executions, go to step 7.
5. ERP must build a Diagnostic Control (F3) command with an Inhibit write subcommand X'02'. The contents of the Program Action Code (Byte 25) defines the path to inhibit as follows:
  - a. If bits 2-7 of sense byte 25 = X'17', set the sub-ID modifier to X'20' (storage controller)
  - b. If bits 2-7 of sense byte 25 = X'18', set the sub-ID modifier to X'40' (Channel Path)
  - c. If bits 2-7 of sense byte 25 = X'19', set the sub-ID modifier to X'80' (Storage Director)
6. Now that the error is permanent for this path Log the I/O error as permanent (even if the Error Condition Table indicated a Record action of No).
7. Do Action 1.

## Action 6

Action 6 does not apply to ESS.



## Action 7

1. Increment the cylinder address of the seek argument in sense bytes 5-6 (or 29-30) by one. Reset the head address to zero. If the incremented cylinder address is still in the current extent, go to step 3. Otherwise, determine the seek address by locating the next extent. If none exists, do Action 2.
2. If a Define Extent command was executed, go to step 4. Otherwise, continue with step 3.
3. Continue the operation by executing the following CCW chain:

**Seek** (argument from step (1))  
**Set File Mask** (same as original)  
**Set Sector** (argument 0)  
**Read Home Address** (skip bit on)  
**TIC** (Interrupted CCW)

4. Continue the operation by executing the following CCW chain:

Define Extent User's new extent compatible with the seek argument from step (1), other parameters are the same as the original

Locate Record Defined as follows:

Byte	Contents
0	'40'
1-3	zeros
4-7	the seek address from step 1
8-11	the seek address from step 1
12-15	zeros
TIC	Interrupted CCW

## Action 8

This is a File Protected error. Examine the interrupted CCW to select step 1, 2, 3, or 4:

1. If the interrupted CCW is a Perform Subsystem Function command, exit to Action 2.
2. If the interrupted CCW is a SEEK, Seek Cylinder, Seek Head, or Recalibrate command:
  - a. If the seek address is still in the current extent, go to the next step. (The seek address for a Recalibrate command is 0.) Otherwise, determine the seek address by locating the next extent. If none exists, do Action 2.
  - b. If a Define Extent command was executed, go to step (d). Otherwise, continue with next step.
  - c. Continue the operation by executing the following CCW chain:

**Seek** argument from step (a)  
**Set File Mask** same as original  
**TIC** The first non-TIC CCW following the interrupted CCW
  - d. Continue the operation by executing the following CCW chain:

Seek argument from step (a)  
Define Extent User's new extent compatible with the seek argument from steps (a), other parameters the same as original  
TIC The first non-TIC CCW following the interrupted CCW
3. If the interrupted CCW was a Locate Record command:

- a. Determine if the seek address is within the user's extents. If not, do Action 2.
  - b. Continue the operation by executing the following CCW chain:
    - Define Extent** User's new extent compatible with the seek argument, other parameters same as original
    - TIC** Interrupted CCW
4. If none of the previous conditions are true, this is either a multi-track CCW or the domain of a locate has gone beyond the current extent. Perform the following recovery actions:
- a. Increment the seek address in sense bytes 5 and 6 by 1.
    - If the incremented seek address is still in the current user extent, go to step (c). Otherwise, determine the seek address by locating the next extent. If none exists, do Action 2.
  - b. If a Define Extent command was executed, go to step (e). Otherwise, continue with the next step.
  - c. Continue the operation by executing the following CCW chain:
    - Seek** argument from steps (a) and (b)
    - Set File Mask** same as original
    - Set Sector** argument 0
    - Read Home Address** skip bit on
    - TIC** Interrupted CCW
    - Note:** If the modified seek argument is not within the user's extent, then IOS must supply correct seek argument before issuing seek. If that is impossible, then IOS must do Action 2.
  - d. Continue the operation by executing the following CCW chain:
    - Note:** If Imprecise Ending is set, the interrupted CCW is not the failing CCW. Byte 3 of sense contains the count of the of the records (or tracks) remaining to be processed. This value is used to rebuild the restart CCW chain.
    - Define Extent** User's new extent is compatible with the seek argument from steps (a) and (b), other parameters are the same as original
    - Locate Record** Defined as follows:
 

Byte	Contents
<b>0</b>	same as original except: <ul style="list-style-type: none"> <li>• If sense byte 3 = '01' and byte 1, bit 7 = '1', then byte 0, bits 2-7 = '16' and byte 1 = 0</li> <li>• If '01', change to '81' See the following notes.</li> </ul>
<b>1</b>	same as original, except as noted for byte 0
<b>2</b>	zero
<b>3</b>	number of CCWs to process from sense byte 3
<b>4-7</b>	the seek address from step (a) and (b)
<b>8-11</b>	the seek address from step (a) and (b)
<b>12-15</b>	zero
<b>TIC</b>	Interrupted CCW

**Notes:**

- a. If one CCW remains and the original Locate Record specifies read count suffixing, then the Locate Record in the restart must specify Read (16) and no read count suffixing.
- b. If the original Locate Record has a write data, then the Locate Record in the restart must have a data orientation.

**Action 9**

1. Print console message "IEA0468I WRITE INHIBITED PATH (cuu,x) ENCOUNTERED" for the operator.
2. Vary path to device off-line and print console message "IEA0469 PATH (cuu,x) HAS BEEN VARIED OFF-LINE" for the operator.
3. If any alternate path to the device is available, repeat the operation using an alternate data path, else do Action 1.

**Action 10**

A Global Command was rejected because the interface was disabled for Global Commands by a Perform Subsystem Function, Set Special Intercept Condition order. The interface on which the command was rejected is no longer disabled. A retry is not rejected for this reason. This recovery action is used as an operating system intercept mechanism by VM and therefore the recovery is operating system dependent.

**Action 11**

An attention was generated and a Unit Check was added to the status because the interface was disabled for Global Commands by a Perform Subsystem Function, Set Special Intercept Condition order. The interface on which the attention and unit check was presented is still disabled. This recovery action is used as an operating system intercept mechanism by VM and therefore the recovery is operating system dependent.

**Error message**

The following error message is included in the operating environment of all ESS users.

**Message 1**

(Used on all hard errors)

1. Message code
2. Error type - Read, Write or Control
3. Unit designation number and head position
4. Channel designation
5. Status and sense bytes sent to the system

The message code should provide reference to system documentation that fully describes message format and content.

The information on the console should be sufficient to allow the system operator to distinguish between probable:

- Operator errors
- User errors (Programming)
- Data Checks (Media Failures)
- Equipment Checks (Hardware Failures)

The information on the console should be sufficient to allow the system operator and CE to identify the failing channel and device.

The console message should contain status and sense bytes sent by the ESS to the system.

### Messages Defined by Sense

If Message to Operator (byte 1, bit 3) is set, the sense data contains a Operator Message Control in byte 24. These messages are shown in Figure 54 on page 179. For compatibility with existing ERPs, a block of messages are also defined by the message code in format 'F' in 24 Byte Compatibility sense data. The sense data for these messages also have codes in byte 28 and Message to Operator is set.

Messages generated to notify the system of specific actions taken by the subsystem are defined as follows. The messages represent the recommended message content. 'I' messages are informational and may roll off the screen; 'E' messages are error messages and should remain displayed until acknowledged.

Figure 59. Messages defined by Sense

Byte 7	Type	Message
04	E	PINNED DATA FOR VVVVV RETRIABLE-CUU-SSID.P-CC S-DD
F1	I	SUBSYSTEM PROCESSING ERROR- CUU-SSID.P
F4	E	SUBSYSTEM STORAGE IS UNUSABLE-CUU-SSID
F7	E	VVVVV TRACK (cch) FORMAT NOT CORRECT FOR DESTAGE - RC=(code) SSID.P-CC S-DD
F9	I	SUBSYSTEM STORAGE REINITIALIZED - RC=(code) SSID.P-S-DD
FA	E	NONVOLATILE STORAGE UNUSABLE-CUU-SSID.P
FB	E	VVVVV (P0 S0) IS PPRC SUSPENDEED - (P1 S1) (code2)-CUU-SSID.P-S-DD
FC	E	ACCESS PROHIBITED ON (type1)-RC = (code) SSID.P-DD (VOLUME VVVVV) (SUBSYSTEM)
FD	I	CACHING STATUS RESET TO DEFAULT -CUU-SSID.P
FE	I	FAST WRITE IS INHIBITED, NVS BATTERY DEAD - CUU-SSID

Figure 60 (Page 1 of 2). Definitions of abbreviations for messages defined by Sense

Abbrev	Definition
SSID	Subsystem Identifier (sense byte 21)
P	Storage Path number (sense byte 27, bits 6-7 )
CUU	System Device Number
Type I	Information, Operators response not required
Type E	Error, Operators response required
S	String address from sense byte 4, bit 2
DD	Device Number (sense byte 4, bits 2-7)
VVVVV	Volume ID, if the Volume ID is unknown "UNKNOWN"

Figure 60 (Page 2 of 2). Definitions of abbreviations for messages defined by Sense

Abbrev	Definition
P0 S0	Primary or Secondary (sense byte 8, bit 0 = 1 = Secondary)
P1 S1	Primary or Secondary (sense byte 8, bit 1 = 1 = Secondary)
cch	Track cylinder and head (sense bytes 29-31)
type1	SUBSYSTEM if byte 8, bits 0-3 = nonzero, otherwise Volume ID
code	Reason code (sense byte 8, bits 0-7)
code2	"INT'N REQUIRED" if byte 8, bits 2-7 = X'0B', or "CACHE UNAVAILABLE" if byte 8, bits 2-7 = X'08', or "NVS UNAVAILABLE" if byte 8, bits 2-7 = X'07'; otherwise, "FAILED-RC=xx", where 'xx' is byte 8, bits 2-7.

### ERP action messages

If the ERP successfully recovers an operation using the alternate path, the following message is required:

IEA466I PATH (CUU,x) PERMANENT I/O ERROR, FAULT CODE=cccc **where:**

- (CUU,x) is the device address and the channel set of that address
- cccc is FCS from sense bytes 22-23.

If an unrecoverable write error detected on a specific path causes the ERP to inhibit write operations on the path indicated by the sense data, the following message is required:

IEA467E PATH (CUU,x) WRITE INHIBITED (type) FOR ALL WRITE OPERATIONS **where:**

- (CUU,x) is the device address and the channel set of that address
- (type) is the component that is Write Inhibited by the Diagnostic Control command, | CHANNEL | SD | CONTROLLER |

If a write command is issued on a path that had been previously 'write inhibited' (Sense data includes Equipment Check (byte 0, bit 3) and Write Inhibited (byte 1, bit 6)), the following message is required.

IEA468I WRITE INHIBITED PATH (CUU,x) ENCOUNTERED **where:**

- (CUU,x) is the device address and the channel set of that address

If the ERP encounters an unrecoverable I/O error on a path and varies that path off-line, the following message is required:

IEA469I PATH (CUU,x) HAS BEEN VARIED OFFLINE **where:**

- (CUU,x) is the device address and the channel set of that address

If the ERP encounters an unrecoverable I/O error on a path and determines that the path should be varied offline, the following message is required:

IEA469I VARY PATH (CUU,x) OFFLINE **where:**

- (CUU,x) is the device address and the channel set of that address

**Note:** This is used in place of the previous message if the system is not capable of automatic vary-off path.

If the ERP encounters an unrecoverable I/O error on the last path to a device and determines that the path should be varied off-line, the following message is required:

IEA469I PATH (CUU,x) CANNOT BE VARIED OFFLINE **where:**

- (CUU,x) is the device address and the channel set of that address

**Note:** This is used if the system ERP cannot vary the path off because it is the last path to the device.

---

## Coded ERP description

Program action numbers used in this section are hexadecimal numbers that denote the binary value presented in sense byte 25.

Logging and message functions indicated by sense byte 24 are to be performed as a part of all program actions. If a recording action is specified, the system must create one of the following records:

- MDR type 90 (Miscellaneous Data Record) for records read from Read and Reset Buffered Log Command.
- MDR type 91 for records presented as format 6 from a unit check
- A3 for SIM records.
- OBR type 30 for all other records.

The retry and recovery actions specified for a particular exception represent the actions deemed by the subsystem designer to be both necessary and sufficient to recover from the condition, if recovery is possible. In general, it is not likely that additional retries can increase the probability of recovery.

Since ESS can neither associate the several channel program executions that comprise a retry sequence, nor distinguish retries from independent I/O operations that are not related to a prior failed operation, it requests the same program action for each occurrence of an exception condition without regard for the number of times the operation has been retried by the host system ERP. It is the responsibility of the ERP to maintain the required counts of the number of retry attempts and terminate the retry sequence after the specified number of attempts. Similarly, the ERP must be prepared to deal with "nested" exception conditions. For example, an equipment check that occurs during a retry sequence that was initiated to recover from a data check.

The ERP is also responsible for generating and presenting appropriate messages to inform the user or system operator of significant actions taken by the ERP (such as recognition of permanent error conditions, disabling one or more paths to a device or subsystem, and so on). In general, the operator message facilities described in "Messages Defined by Sense" on page 210 are used only to provide subsystem-defined messages to the system operator. These messages are typically used to report events or conditions that are detected by the subsystem, and which may not be apparent to a system program.

## Coded program actions

Error recovery program actions are indicated by the Program Actions Codes (PAC) in sense data byte 25 and can be in one of two formats:

1. An action that is composed of one or more common error recovery procedures (Compound Program Action Codes)
2. A unique error recovery action designed for one specific exception condition (Single Program Action Codes).

Sense data byte 25, bit 0 specifies which PAC format is to be used to interpret the exception condition:

### Bit Description

**0:** Format Indicator

#### Value

- 1** Bits 1-7 Contain Compound PAC
- 0** Bits 1-7 Contain Single PAC

When bit 0 = '1', bits 1-7 are indicators that identify the required common error recovery program actions or point to other sense data bytes that identify unique error recovery actions. This format is used when the error recovery action is composed of one or more common actions that may be used in various combinations. These Compound Program Action Codes are defined in "Compound Program Action Codes."

When bit 0 = '0', bits 1-7 represent a hexadecimal value that identifies a Program Action Code (PAC) that is unique to the specific exception condition. These Single Program Action Codes are defined in "Single Program Action Codes" on page 215.

## Compound Program Action Codes

Compound Program Action Codes defined in this section are bit indicators specifying common error recovery program actions and a retry count. These are present in byte 25, bits 1-7 when byte 25, bit 0 = '1'.

### Bit Description

**1** Configuration Dependent Function

#### Value

- 0** No function requested
- 1** See bytes 26-27

**2** Message code

#### Value

- 0** No function requested
- 1** See byte 28

**3** Alternate Path

#### Value

- 0** No Alternate Path retry
- 1** Alternate Path retry

**4-5** Reserved

**6-7** Retry count

#### Value

- 00** No Retry
- 01** 2 times
- 10** 10 times
- 11** 255 times

**Byte 25, bit 1** when '1', this bit is always set to zero by ESS.

**When byte 25, bit 2 = '1'**, specifies that sense data byte 28 contains a message code that identifies which area to 'Write Inhibit' if all retries are exhausted for the channel path.

The error codes in byte 28 are:

**Code Action**

- 17** If the error condition persists on this path, build a Diagnostic Control (F3) command with an Inhibit write subcommand X'02' and a subcommand modifier of X'20' (Controller).
- 25** This error may take up to 5 seconds before accepting a redrive. Users with high performance requirements and an alternate data source should obtain the data from the alternate source.

When bit 2 = '0', no error recovery program action is specified for any message code present in byte 28.

**When byte 25, bit 3 = '1'**, retry the operation on each additional channel path to the device, other than the channel path on which the exception condition occurred. If the exception condition persists on all paths, execute any other error recovery action indicated. If no other error recovery action is indicated, exit with a permanent error indication.

When bit 3 = '0', do not retry the operation on additional channel paths.

**Byte 25, bits 6-7**, specifies the number of times to retry an operation using the same channel path on which the exception condition occurred:

- When bits 6-7 = '00', do not retry the operation on this channel path.
- When bits 6-7 = '01', retry the operation two times on this channel path.
- When bits 6-7 = '10', retry the operation ten times on this channel path.
- When bits 6-7 = '11', retry the operation 255 times on this channel path.

If the exception condition persists and bit 3 = '0', execute any other recovery action indicated. If no other error recovery action is indicated, exit with a permanent error indication.

If the exception condition persists and bit 3 = '1', attempt the operation on each additional channel path until the operation is completed successfully or all channel paths have been tried. If the exception condition persists on all channel paths, execute any other error recovery action indicated. If no other error recovery action is indicated, exit with a permanent error indication.

**Programming Note:** The retry and recovery actions specified for a particular exception represent the actions deemed by the subsystem designer to be both necessary and sufficient to effect recovery from the condition, if recovery is possible. In general, it is not likely that additional retries increase the probability of recovery.

Since the I/O subsystem can neither associate the several channel program executions that comprise a retry sequence, nor distinguish retries from I/O operations that are not related to a prior failed operation, it requests the same program action for each occurrence of an exception condition without



regard for the number of times the operation has been retried by the host system Error Recovery Program (ERP). The ERP must maintain a count of the number of retry attempts and terminate the retry sequence after the specified number of attempts.

The ERP must also be prepared to deal with “nested” exception conditions as, for example, an Equipment Check that occurs during a retry sequence that was initiated to recover from a Data Check.

The ERP should be prepared to generate and present appropriate messages to inform the user or system operator of significant actions taken by the ERP (such as recognition of permanent error conditions, disabling one or more paths to a device or subsystem, and so on).

**Programming Note:** An exception condition may occur that makes it inadvisable to attempt further write operations using a specific facility.

If the exception condition persists through all retry actions, the error recovery program should perform the model-dependent procedure described in the control unit functional specifications to inhibit all further write operations on the facility.

This action, known as Inhibit Write, is requested when the failure makes it likely that further attempts to perform write operations could result in permanent errors and the possible destruction of data.

The rationale for requesting the retries on the same path that failed includes the following:

1. There may be data paths internal to the subsystem and not visible to the ERP that are used during the retry attempts
2. Retrying on the same path verifies that the error is persistent on that path.

### Single Program Action Codes

Program Action Codes (PAC) used in this section are hexadecimal numbers that represent the binary value presented in sense byte 25, bits 1-7 when byte 25, bit 0 = '0'.

**Note:** Restart Channel Programs defined in the following “Actions” may include a Locate Record command. This command can be either a Locate Record or Locate Record Extended, depending on the command used in the original channel program.

When an I/O exception occurs within the domain of a Locate Record Extended command, the error recovery program must determine from the contents of byte 19 whether an Extended Parameter was present in the original command. If the recovery action involves the length or content of an Extended Parameter, the program must obtain this data from the original command within processor storage since this information is not available in the sense data.

Error recovery programs may be unable to find the original Locate Record Extended command when the command follows a part of a channel program that contains conditional branching (via Status Modifier). In this case, no recovery action may be possible.

**Action 00:** No program action is required; exit with no error indication.

**Actions 01:** Retry is not recommended. Exit with a Permanent Error.

**Action 02:** An intervention required condition has been detected on the reporting unit.

Issue a message to the operator indicating that intervention is required on the reporting device, then queue the I/O request until an interrupt is received from the device indicating the device has been made ready.

**Actions 03-0F:** Reserved

**Action 10:** The condition that caused the Unit Check to be presented is not related to the channel program that is being initiated. After performing any logging and message action indicated in sense byte 24, bits 4-7, restart the channel program.

If the exception condition persists through 255 retries, exit with a permanent error indication.

**Note:** Normally, the first restart of the channel program is successful. A retry count greater than one is required to handle situations where the subsystem has many pending unit check or subsystem information conditions to present. The limit of 255 on the count is imposed to cause a permanent error indication in case a failure in the subsystem causes the same condition to be reported repeatedly.

**Action 11 through 14:** Reserved

**Action 15:** A multi-track data transfer operation within the domain of a Locate Record command has attempted to switch to a new track and the next track is outside the defined extent.

Sense bytes 29-31 contain the address 'CCH' of the last track on which the subsystem operated.

Determine the address of the first track in the user's next extent. If there is no other extent, exit with a permanent error indication; otherwise, determine the address of the CCW to be restarted using the count of the completed operations and the command address in the SCSW.

Then restart the interrupted channel program with the following:

Define Extent:       Extent boundaries compatible with the seek address developed in above paragraph.

Locate Record as follows:

Byte	Contents
0	from sense byte 7, except: <ul style="list-style-type: none"><li>• If sense byte 3 = '01' and byte 1, bit 7 = '1', then byte 0, bits 2-7 = '16' and byte 1 = '0'</li><li>• If '01', change to '81'</li></ul> See the following notes.
1	from sense byte 8, except as noted for byte 0
2-3	from sense byte 9 and 10
4-7	Seek address from paragraph above
8-12	Search Argument consistent with the operation code and seek address
13-15	from sense bytes 16-18
TIC	To the CCW to be re-started

**Note:** If one CCW remains and the original Locate Record specifies read count suffixing, then the Locate Record in the restart must specify Read (16) and no read count suffixing.

**Note:** If the original Locate Record has a write data, then the Locate Record in the restart must have a data orientation.

**Action 16 through 1B:** Reserved

**Action 1C:** While executing a channel program in PCI fetch mode, the subsystem performed a retry action.

The validity of the data transferred to the channel may be suspect and the user should take action if the data were accessed before the channel program completed.

**Action 1D:** A state change pending condition exists for the subsystem or device.

**Actions 1E-FF:** Reserved

**Note:** X'70 and 71' are used in 24 Byte Compatibility sense, but not in ECKD 32 Byte sense.



---

## Appendix A. Device data, geometries, and facilities

This chapter contains the following tables:

- Figure 61 on page 220 contains values returned by Read Device Characteristics.
- Figure 62 on page 221 contains values returned by Sense ID.
- Figure 63 on page 221 contains device geometries and facilities.
- Figure 64 on page 222 lists device track ranges, and contains:
  - Device track Group Address Ranges
  - Status track address
  - Access Authorization address Ranges.

Figure 61 (Page 1 of 2). Device Characteristics. The specific value of the device characteristics information transferred by Read Device Characteristics (64) for devices attached to the subsystem is described below.

All device characteristics information in the following tables is represented in hexadecimal; byte displacements are in decimal. See "Read Device Characteristics" on page 54 for additional definition of the bytes.

Bytes	Description	3390-2		3390-3		3390-9
		Nat	Emul	Nat	Emul	Nat
0-1	storage facility type	<b>Emulation</b> 3990 ESS		<b>Type Code</b> 3990 2105		
2	CU Model	<b>Emulation</b> 3990 (Basic Operation Mode) 3990 (3990-6 Enhanced Operating Mode 1) ESS		<b>Model Code</b> EC E9 E8		
3-4	Device type	3390	3380	3390	3380	3390
5	Device model	06	8A	0A	9E	0C
6-9	Device and SD facilities	See description in section "Read Device Characteristics" on page 54 to determine which values are set.				
10	Device Class Code	20	20	20	20	20
11	Device type code	27	0E	24	0E	32
12-13	Primary cylinders	08B2	08B2	0D0B <sup>3</sup>	0D0B <sup>3</sup>	2721 <sup>3</sup>
14-15	Tracks per cylinder	000F	000F	000F	000F	000F
16	Number of sectors	E0	DE	E0	DE	E0
17-19	Track length	00E5A2	00BB60	00E5A2	00BB60	00E5A2
20-21	Length of HA and R0	0594	0440	0594	0440	0594
22	Track capacity formula	02	01	02	01	02
23	Factor F1	22	20	22	20	22
24-25	Factor F2 (F2F3)	(1309)	01EC	(1309)	01EC	(1309)
26-27	Factor F3 (F4F5)	(0674)	00EC	(0674)	00EC	(0674)
28-29	First alternate cylinder	0	0	0	0	0
30-31	Number of alternate trks	0	0	0	0	0
32-33	First diagnostic cyl	0	0	0	0	0
34-35	Number of diag. tracks	0	0	0	0	0
36-37	First device support cyl	0	0	0	0	0
38-39	Number of DS tracks	0	0	0	0	0
40	MDR record ID	27	27	24	24	32
41	OBR record ID	27	27	24	24	32
42	CU Type Code	<b>Emulation</b> 3990 (Basic Operation Mode) 3990. (3990-6 Enhanced Operating Mode 1) ESS		<b>Type Code</b> 06 15 1B		
43	Parameter Length	2	2	2	2	2
44-45	Record 0 Length	DFEE	BB74	DFEE	BB74	DFEE
46	Unassigned	zeros				
47	Track Set	01	01	01	01	01
48	F6	06	00	06	00	06
49-50	RPS Factor	7708	5007	7708	5007	7708
51-53	Reserved					
54	Generic Device/CU Functions/Features	See description in section "Read Device Characteristics" on page 54 to determine which values are set.				
55	Reserved	00				
56	Real CU Type Code	<b>Emulation</b> 3990 ESS	<b>Type Code</b> 1B 00			
57	Real Device OBR Code	<b>Model</b> CPSS Drawer	<b>Type Code</b> 35			

Figure 61 (Page 2 of 2). Device Characteristics. The specific value of the device characteristics information transferred by Read Device Characteristics (64) for devices attached to the subsystem is described below.

All device characteristics information in the following tables is represented in hexadecimal; byte displacements are in decimal. See "Read Device Characteristics" on page 54 for additional definition of the bytes.

Bytes	Description	3390-2		3390-3		3390-9
		Nat	Emul	Nat	Emul	Nat
58-63	Unassigned	zeros				

<sup>2</sup> See Section "Read Device Characteristics" on page 54 for specific values.

<sup>3</sup> Value may be smaller if this is a Flex-Volume.

Figure 62. Sense ID Values. The specific value of the Sense ID information transferred by Sense ID (E4) for devices attached to the subsystem is described below.

All device characteristics information in the following tables is represented in hexadecimal; byte displacements are in decimal.

Bytes	Description	3390-2		3390-3		3390-9
		Nat	Emul	Nat	Emul	Nat
0		FF				
1-2	Storage Control Type	<b>Emulation</b> <b>3990</b> <b>ESS</b>		<b>Type Code</b> 3990 2105		
3	CU Model	<b>Emulation</b> <b>3990 (Basic Operation Mode)</b> <b>3990 (3990-6 Enhanced Operating Mode 1)</b> <b>ESS</b>		<b>Model Code</b> EC E9 E8		
4-5	Device type	3390	3380	3390	3380	3390
6	Device model	06	8A	0A	9E	0C
7	Reserved	00				
8-12	CIW for RCD	40FA0100				
12-15	CIW for SII	41270004				
16-19	CIW for RNI	423E01A0				
20-23	CIW for RAM	433E0008				

Figure 63. Device Geometry and Facilities. The device specific limits and sizes are given below in decimal.

Description	3390-2		3390-3		3390-9
	Nat	Emul	Nat	Emul	Nat
Usable track capacity (bytes)	57326	47988	57326	47988	57326
Maximum record length (bytes)	56664	47476	56664	47476	56664
Data Cylinders per actuator	2226	2226	3339 <sup>3</sup>	3339 <sup>3</sup>	10017 <sup>3</sup>
Tracks per Cylinder	15	15	15	15	15
Capacity/actuator (mbytes)	1892	1585	2838 <sup>3</sup>	2378 <sup>3</sup>	8514 <sup>3</sup>
Sectors per Track	224	222	224	222	224
Range of Sector numbers	0-223	0-221	0-223	0-221	0-223
RPS Facility	yes	yes	yes	yes	yes
Def/Alt Track Facility	no	no	no	no	no
Diagnostic Track Facility	no	no	no	no	no
Device Support Track Facility	no	no	no	no	no

<sup>3</sup> Value may be smaller if this is a Flex-Volume.

Figure 64. Device Track Address Ranges. The device specific track address ranges are given below in hexadecimal.

Description		3390-2	3390-3	3390-9
<b>Device Track Group Address Range:</b>				
Primary Track address range	Low	000000	000000	000000
	High	08B10E	0D0B0E	27200E
<b>Access Authorization Address Range:</b> The track address ranges allowed in a Define Extent command or a Seek command for the various Access Authorizations are given below, by device, in hexadecimal in the form 'CC' (2 bytes) for the cylinder address ranges and 'H' (1 byte) for the head address ranges. Invalid addresses within a range are not valid as extent or seek addresses but may be included within the extent range.				
Normal Access address range	Low	000000	000000	000000
	High	08B10E	0DB00E	27200E
Device Support address ranges	Low	000000	000000	000000
	High	08B10E	0DB00E	27200E
Diagnostic address range	Low	Not Valid	Not Valid	Not Valid
	High	Not Valid	Not Valid	Not Valid



---

## Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. 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 IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described 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:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publications. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

### Read Before Using

#### IMPORTANT

YOU ACCEPT THE TERMS OF THIS IBM LICENSE AGREEMENT FOR MACHINE CODE BY YOUR USE OF THE HARDWARE PRODUCT OR MACHINE CODE. PLEASE READ THE AGREEMENT CONTAINED IN THIS BOOK BEFORE USING THE HARDWARE PRODUCT. SEE "IBM Agreement for Licensed Internal Code" on page 229.



---

## Trademarks

The following terms are trademarks of the IBM Corporation in the United States, other countries or both:

AIX  
Application System 400  
AS/400  
Enterprise Storage Server  
ESCON  
IBM  
MVS  
Netfinity  
OS/390  
OS/400  
RISC System/6000  
RS/6000  
RISC System/6000 Series Parallel  
RS/6000 SP  
S/370  
S/390  
Seascape  
StorWatch  
System/390

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

Other company, product, and service names, may be trademarks or service marks of others.

---

## Electronic emission notices

This section gives the electronic emission notices or statements for the United States and other countries.

---

## Federal Communications Commission (FCC) statement

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Properly shielded and grounded cables and connectors must be used in order to meet FCC emission limits. IBM is not responsible for any radio or television interference caused by using other than recommended cables and connectors or by unauthorized changes or modifications to this equipment. Unauthorized changes or modifications could void the user's authority to operate the equipment.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

---

## Industry Canada compliance statement

This Class A digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

---

## European Community compliance statement

This product is in conformity with the protection requirements of EC Council Directive 89/336/EEC on the approximation of the laws of the Member States relating to electromagnetic compatibility. IBM cannot accept responsibility for any failure to satisfy the protection requirements resulting from a non-recommended modification of the product, including the fitting of non-IBM option cards.

---

## Germany only

### **Zulassungsbescheinigung laut Gesetz ueber die elektromagnetische Vertraeglichkeit von Geraeten (EMVG) vom 30. August 1995.**

Dieses Geraet ist berechtigt, in Uebereinstimmung mit dem deutschen EMVG das EG-Konformitaetszeichen - CE - zu fuehren.

Der Aussteller der Konformitaetserklaeung ist die IBM Deutschland.

Informationen in Hinsicht EMVG Paragraph 3 Abs. (2) 2:

Das Geraet erfuellt die Schutzanforderungen nach EN 50082-1 und EN 55022 Klasse A.

EN 55022 Klasse A Geraete beduerfen folgender Hinweise:

Nach dem EMVG:

"Geraete duerfen an Orten, fuer die sie nicht ausreichend entstoert sind, nur mit besonderer Genehmigung des Bundesministeriums fuer Post und Telekommunikation oder des Bundesamtes fuer Post und Telekommunikation betrieben werden. Die Genehmigung wird erteilt, wenn keine elektromagnetischen Stoerungen zu erwarten sind." (Auszug aus dem EMVG, Paragraph 3, Abs.4)

Dieses Genehmigungsverfahren ist nach Paragraph 9 EMVG in Verbindung mit der entsprechenden Kostenverordnung (Amtsblatt 14/93) kostenpflichtig.

Nach der EN 55022:

"Dies ist eine Einrichtung der Klasse A. Diese Einrichtung kann im Wohnbereich Funkstoerungen verursachen. in diesem Fall kann vom Betreiber verlangt werden, angemessene Massnahmen durchzufuehren und dafuer aufzukommen."

Anmerkung:

Um die Einhaltung des EMVG sicherzustellen, sind die Geraete wie in den Handbuechern angegeben zu installieren und zu betreiben.

---

## Japanese Voluntary Control Council for Interference (VCCI) Class

A statement

この装置は、情報処理装置等電波障害自主規制協議会（VCCI）の基準に基づくクラス A 情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

VS1KAN

---

## Korean Government Ministry of Communication (MOC) statement

Please note that this device has been approved for business purpose with regard to electromagnetic interference. If you find this is not suitable for your use, you may exchange it for a non-business purpose one.

---

## Taiwan Class A Compliance statement

**警告使用者:**

這是甲類的資訊產品，在居住的環境中使用時，可能會造成射頻干擾，在這種情況下，使用者會被要求採取某些適當的對策。



---

## IBM Agreement for Licensed Internal Code

You accept the terms of this Agreement<sup>5</sup>

by your initial use of a machine that contains IBM Licensed Internal Code (called "Code"). These terms apply to Code used by certain machines IBM or your reseller specifies (called "Specific Machines"). International Business Machines Corporation or one of its subsidiaries ("IBM") owns copyrights in Code or has the right to license Code. IBM or a third party owns all copies of Code, including all copies made from them.

If you are the rightful possessor of a Specific Machine, IBM grants you a license to use the Code (or any replacement IBM provides) on, or in conjunction with, only the Specific Machine for which the Code is provided. IBM licenses the Code to only one rightful possessor at a time.

Under each license, IBM authorizes you to do only the following:

1. Execute the Code to enable the Specific Machine to function according to its Official Published Specifications (called "Specifications");
2. Make a backup or archival copy of the Code (unless IBM makes one available for your use), provided you reproduce the copyright notice and any other legend of ownership on the copy. You may use the copy only to replace the original, when necessary; and
3. Execute and display the Code as necessary to maintain the Specific Machine.

You agree to acquire any replacement for, or additional copy of, Code directly from IBM in accordance with IBM's standard policies and practices. You also agree to use that Code under these terms.

You may transfer possession of the Code to another party only with the transfer of the Specific Machine. If you do so, you must 1) destroy all your copies of the Code that were not provided by IBM, 2) either give the other party all your IBM-provided copies of the Code or destroy them, and 3) notify the other party of these terms. IBM licenses the other party when it accepts these terms. These terms apply to all Code you acquire from any source.

Your license terminates when you no longer rightfully possess the Specific Machine.

---

## Actions You Must Not Take

You agree to use the Code only as authorized above. You must not do, for example, any of the following:

1. Otherwise copy, display, transfer, adapt, modify, or distribute the Code (electronically or otherwise), except as IBM may authorize in the Specific Machine's Specifications or in writing to you;
2. Reverse assemble, reverse compile, or otherwise translate the Code unless expressly permitted by applicable law without the possibility of contractual waiver;
3. Sublicense or assign the license for the Code; or
4. Lease the Code or any copy of it.

---

<sup>5</sup> Form Z125-4144.





---

# Glossary

## A

**allegiance.** The ESA/390 term for a relationship that is created between a device and one or more channel paths during the processing of certain conditions. Refer to implicit allegiance, contingent allegiance, and reserved allegiance.

**ANSI.** American National Standards Institute. A standards committee body for the United States.

**ASCII.** A coding scheme defined by ANSI X3.4-1977 which is used to represent various alphabetic, numeric, and special symbols with a seven-bit code.

## B

**bit.** binary digit. The storage medium required to store a single binary digit.

**byte.** An aggregation of eight bits. The storage medium required to store eight bits.

**block.** A group of consecutive bytes.

## C

**cache fast write data.** Data that has the attribute that it can easily be recreated by the control program in the event it is lost. Cache fast write data may be cached in volatile memory.

**cache memory.** Typically refers to volatile memory used to improve access times on data or instructions accesses. The cache memory is typically smaller and faster than the data access storage medium. In a storage facility, the cache memory is used to improve performance.

**CCR.** channel command retry.

**CCW.** channel command word. The ESA/390 term for a data structure that is used to specify an I/O operation to the channel subsystem.

**CEC.** computer electronic complex. The set of hardware facilities associated with a host computer.

**channel.** The ESA/390 term for the part of a channel subsystem that manages a single I/O interface between a channel subsystem and a set of control units.

**channel-command retry.** An ESA/390 term for a protocol used between a channel and a control unit

which allows the control unit to request that the current command be reissued.

**channel path.** The ESA/390 term for the interconnection between a channel and its associated control units.

**channel subsystem.** The ESA/390 term for the part of a host computer that manages I/O communication between the program and any attached control units.

**channel-subsystem image.** The ESA/390 term for the logical facilities required to perform the function of a channel subsystem. With EMIF, one channel-subsystem image exists in the channel subsystem for each LPAR. Each image appears to be an independent channel subsystem program, but all images share a common set of hardware facilities.

**CKD.** count key data. An ESA/390 architecture for a DASD logical device that specifies the format of and access mechanisms for the logical data units on the device. The logical data unit is a track which can contain one or more records, each consisting of a count field, a key field (optional), and a data field (optional).

**cluster.** A partition of a storage server which is capable of performing all defined functions of the storage server. Multiple clusters provide redundancy.

**Concurrent Copy.** A facility of a storage server used to assist a program in making a consistent copy of a logical volume while the logical volume remains available for subsequent processing.

**consistent copy.** A copy of a data entity (e.g. a logical volume) that contains the contents of the entire data entity from a single instant in time.

**contingent allegiance.** The ESA/390 term for a relationship that is created in a control unit between a device and a channel path when unit-check status is accepted by the channel. The allegiance causes the control unit to guarantee access (i.e. busy status is not presented) to the device to retrieve the sense data associated with the unit-check status on the channel path associated with the allegiance.

**control unit.** The ESA/390 term for a storage server with ESCON interfaces.

**control-unit image.** The ESA/390 term for a logical subsystem accessed via an ESCON I/O Interface. One or more control-unit images exist in each control unit. Each image appears to be an independent control unit, but all images share a common set of hardware facilities.

**count field.** The first field of a CKD record. The eight-byte field contains a four-byte track address (CCHH) that defines the cylinder and head associated with the track, a one-byte record number (R) that identifies the record on the track, a one-byte key length that specifies the length of the record's key field (0 means no key field), and a two-byte data length that specifies the length of the record's data field (0 means no data field). Only the end-of-file record has a data length of zero.

**CPAR.** customer problem analysis and repair.

**CPSS.** Common Platform Storage Subsystems. A strategy and family of storage products based on common hardware parts.

**CRC.** cyclic redundancy check.

**CSU.** customer set up.

**cylinder.** A unit of storage on a CKD device consisting of a fixed number of tracks.

## D

**DA.** device adapter.

**DASD.** Direct Access Storage Device.

**DASD Fast Write.** A facility of a storage server that allows caching of active write data without exposure of data loss by journaling of the active write data in NVS.

**data field.** The third (optional) field of a CKD record. The field length is determined by the data length specified in the count field. The data field contains data written by the program.

**data record.** A data records are stored on a track following the track-descriptor record. The data records are numbered consecutively starting with 1. A maximum of 255 data records can be stored on a track. Each data record consists of a count field, a key field (optional), and a data field (optional).

**demote.** The action of removing a logical data unit from cache memory. A data unit is demoted in order to make room for other logical data units in the cache or because the logical data unit was invalidated. Logical data units with active write must be destaged before being demoted.

**destage.** Destage is the process of moving data that has been modified in cache to the DDMS.

**device.** The ESA/390 term for a logical device.

**device adapter.** A physical sub-unit of a storage server that provides the facilities to attach to one or

more interfaces used to communicate with the associated storage devices.

**device address.** The ESA/390 term for the field of an ESCON device-level frame that selects a specific device on a control-unit image.

**device number.** The ESA/390 term for a four-hexadecimal-character identifier (e.g. X'13A0') associated with a device to facilitate communication between the program and the host operator. The device number is associated with a subchannel.

## E

**EBCDIC.** A coding scheme developed by IBM which is used to represent various alphabetic, numeric, and special symbols with an 8 bit code.

**EC.** engineering change. A update to a machine, part, or program. Each EC for a given unit is assigned a unique number referred to as an EC level or EC number. The initial release of a unit normally has an associated EC level.

**ECC.** error correction code.

**ECKD.** extended count key data. An extension of the CKD architecture.

**EMIF.** ESCON Multiple Image Facility. An ESA/390 facility that allows LPARs to share an ESCON channel path by providing each LPAR with its own channel-subsystem image.

**end of file.** On a CKD DASD device, end of file is indicated by a record with a data length of zero.

**ERP.** error recovery procedure.

**ESA/390.** Enterprise Systems Architecture/390 An IBM architecture for mainframe computers and peripherals. Processor systems following this architecture include the ES/9000 family.

**ESCON.** Enterprise Systems Connection Architecture. An ESA/390 computer peripheral interface. The I/O interface utilizes ESA/390 logical protocols over a serial interface that configures attached units to a communication fabric.

**ESCON director.** A I/O interface switch that allows the interconnection of multiple ESCON interfaces in a distributed-star topology.

**Extended Remote Copy.** A facility of a storage server that assists a control program to maintain a consistent copy of a logical volume on another storage facility. All modifications of the primary logical volume by any attached host are presented in order to a single host

which in turn makes these modifications on the secondary logical volume.

## F

**FIFO.** first-in first-out.

**FRU.** field-replaceable unit.

## H

**HA.** home address, host adapter.

**HDA.** head and disk assembly. The portion of a HDD associated with the medium and the read/write head.

**HDD.** head and disk device. A disk drive.

**home address.** A nine-byte field at the beginning of a track that contains information identifying the physical track and its association with a cylinder.

**host.** A computer.

**host adapter.** A physical sub-unit of a storage server that provides the facilities to attach to one or more host I/O interfaces.

## I

**I/O.** input/output.

**I/O interface.** An interface defined to allow input/output operations to be performed between a host and its associated peripheral devices.

**IML.** initial microcode load

**implicit allegiance.** The ESA/390 term for a relationship that is created in a control unit between a device and a channel path, or path group, when an I/O operation is accepted by the device. The allegiance causes the control unit to guarantee access (i.e. busy status is not presented) to the device for the remainder of the channel program over the set of channel paths associated with the allegiance.

**interchange.** The capability of processing (read or write) a given medium on any one of a set of devices which support the form factor and recording format on the medium.

**invalidate.** The action of removing a logical data unit from cache memory because of an inability to support continued access to the logical data unit on the logical device. The inability may be the result of a failure within the storage server or a storage device associated with the logical device.

**IPL.** initial program load

## K

**key field.** The second (optional) field of a CKD record. The field length is determined by the key length specified in the count field. The key field contains data written by the program that can be used to identify or locate a given record.

**KPOH.** thousands of power-on hours. A unit of time used for a measurement of MTBF.

## L

**LIC.** licensed internal code

**licensed internal code.** Firmware, microcode, or software installed on a product that is required by the product to perform its defined function. Licensed internal code is provided by the product vendor with the product.

**LIFO.** last-in first-out

**link address.** On an ESCON interface, the portion of a source or destination address in a frame used route a frame through an ESCON director. The link address is associated with a specific switch port on the ESCON director, or equivalently, with the channel-subsystem or control-unit link-level facility attached to that switch port.

**link-level facility.** An ESCON term for the hardware and logical facilities of a control unit or channel subsystem required to allow communication over an ESCON I/O interface.

**logical address.** On an ESCON interface, the portion of a source or destination address in a frame used to select a specific channel-subsystem or control-unit image.

**logical data unit.** A unit of storage which is accessible on a given logical device.

**logical device.** The facilities of a logical subsystem that the host communicates with when performing I/O operations to a single addressable-unit over an I/O interface. The same logical device may be accessible over more than one I/O interface.

**logical subsystem.** The logical facilities in a storage server required to allow one or more host I/O interfaces to access a set of logical devices that are considered to be aggregated according to the addressing mechanisms of the associated I/O interfaces. One or more logical subsystems exist on a storage server. In general, a given set of logical devices is associated with one and only one logical subsystem.

**logical volume.** The storage medium associated with a logical device. A logical volume typically resides on one or more storage devices.

**LPAR.** logical partition. The ESA/390 term for a set of facilities that create the programming environment defined by the ESA/390 architecture. The term is typically used when there is more than one LPAR established on a CEC. An LPAR is conceptually similar to a virtual machine environment except that the LPAR is a function of the CEC and is not dependent on an operating system to create the virtual machine environment.

**LRU.** least recently used. A policy for a caching algorithm which chooses to remove the item from cache which has the longest elapsed time since its last access.

## M

**MLC.** machine-level control. A data bases that maintains the EC level and configuration of products in the field.

**MTBF.** mean time between failures. A projection of the time that an individual unit will remain functional based on averaging the performance or projected performance of a population of statistically independent units under as set of operating conditions or assumptions.

## N

**ND.** node descriptor.

**NED.** node-element descriptor.

**NEQ.** node-element qualifier.

**non-removable medium.** Recording media that cannot be added to or removed from a storage device.

**non-volatile storage.** Typically refers to non-volatile memory on a processor rather than a non-volatile storage device. On a storage facility, non-volatile storage is used to store active write data to avoid data loss in the event of a power loss.

**NQ.** node qualifier.

**NVS.** non-volatile storage.

## P

**path group.** The ESA/390 term for a set of channel paths that are defined to a control unit as being associated with a single LPAR and as being in the grouped state (e.g. online to the host).

**path-group identifier.** The ESA/390 term for the identifier that uniquely identifies a given LPAR. The path-group identifier is used in communication between the LPAR program and a device to associate the path-group identifier with one or more channel paths, thereby defining these paths to the control unit as being associated with the same LPAR.

**PCI.** program-controlled interrupt.

**Peer to Peer Remote Copy.** A facility of a storage server that maintains a consistent copy of a logical volume on the same storage facility or on another storage facility that is accessible over an ESCON I/O interface. All modifications of the primary logical volume by any attached host are also performed in order by the storage facility on the secondary logical volume. Options exist to request the modifications to be performed synchronously (i.e. before the completion of the modification of the primary copy is are completed).

**PPRC.** Peer to Peer Remote Copy.

**predictable write.** A write operation that can cached without knowledge of the existing formatting on the medium. All writes on FBA DASD devices are predictable. On CKD DASD devices, a write is predictable if it does a format write for the first data record on the track.

**program.** A generic term for the software that controls the operation of a host computer. Typically, the program is an operating system that allows sharing of the host resources between multiple tasks.

**promote.** The action of adding a logical data unit to cache memory from a device.

## R

**random access.** A mode of accessing data on a medium in a manner that requires the storage device to access non-consecutive storage locations on the medium.

**reserved allegiance.** The ESA/390 term for a relationship that is created in a control unit between a device and a channel path, or path group, when a Sense Reserve command is completed by the device. The allegiance causes the control unit to guarantee access (i.e. busy status is not presented) to the device over the set of channel path associated with the

allegiance, for one or more channel programs, until the allegiance is terminated.

**R0.** track-descriptor record.

## S

**SBCON.** Single-Byte Command Code Sets Connection Architecture. An ANSI standard for the ESCON I/O interface.

**sequential access.** A mode of accessing data on a medium in a manner that requires the storage device to access consecutive storage locations on the medium.

**Sequential Stage Group.** The number of tracks that will be prestaged in a sequential operation.

**service processor.** A dedicated processing unit which is used to service a storage facility.

**SIM.** service-information message.

**SSA.** Serial Storage Architecture. An IBM standard for a computer peripheral interface. The interface utilizes a SCSI logical protocol over a serial interface that configures attached targets and initiators in a ring topology.

**SSID.** subsystem identifier.

**stacked status.** An ESA/390 term for status that the control unit is holding for the channel because the channel responded with the stack-status control the last time the control unit attempted to present the status.

**STI.** self-timed interface. An interface that consists of one or more conductors that transmit information serially between two interconnected units without requiring any clock signals to recover the data. Clock recovery is performed independently on each serial data stream and information in the data stream is used to determine character boundaries and inter-conductor synchronization.

**storage device.** A physical unit which provides a mechanism to store data on a given medium such that it can be subsequently retrieved.

**storage facility.** A physical unit which consists of a storage server integrated with one or more storage devices to provide storage capability to a host computer.

**storage server.** A physical unit which provides an interface between one or more storage devices and a host computer by providing the function of one or more logical subsystems. The storage server may provide

functions that are not provided by the storage device. The storage server is composed of one or more clusters.

**subchannel.** A logical facilities of a channel subsystem associated with the management of a single device.

**subsystem identifier.** A number that uniquely identifies a logical subsystem within a computer installation.

**synchronous write.** A write operation whose completion is indicated after the data has been stored on a storage device.

## T

**track.** A unit of storage on a CKD device consisting that can be formatted to contain a number of data records. Also see *home address*, *track-descriptor record*, and *data record*.

**track-descriptor record.** A special record on a track following the home address that is used by the control program to maintain certain information about the track. The record has a count field with a key length of zero, a data length of 8, and a record number of 0. This record is sometimes referred to as R0.

## U

**unit address.** The ESA/390 term for the address associated with a device on a given control unit. On ESCON interfaces, the unit address is the same as the device address. On OEMI interfaces, the unit address specifies a control unit/device pair on the interface.

**utility device.** The ESA/390 term for the device used with the Extended Remote Copy facility to access information describing the modifications performed on the primary copy.

## V

**volume.** The ESA/390 term for the set of information recorded on a single unit of recording medium or, indirectly, refers to the unit of recording medium itself. On a non-removable medium storage device, the terms may also refer, indirectly to the storage device associated with the volume. When multiple volumes are stored on a single storage medium transparently to the program, the volumes may be referred to as logical volumes.

**VPD.** vital product data entered via ESS Specialist.

# **X**

**XRC.** Extended Remote Copy.

---

# Index

## Numerics

- 3390-1
  - device characteristics 219
  - device facilities 221
  - device geometry 221
  - SNSID values 221
- 3390-2
  - device characteristics 219
  - device facilities 221
  - device geometry 221
  - SNSID values 221
- 3390-3
  - device characteristics 219
  - device facilities 221
  - device geometry 221
  - SNSID values 221
- 3390-9
  - device characteristics 219
  - device facilities 221
  - device geometry 221
  - SNSID values 221

## A

- access
  - authorization 13
    - device support 24, 14, 29, 36
    - diagnostic 24, 61, 75, 92, 93, 94
    - normal 24, 14
  - data 14
- accessing
  - record 16
  - track 14
- accessing ESS Specialist and ESS Copy Services 5
- Activate Cache Fast Write Access for Subsystem 110
- agreement for Licensed Internal Code 229
- area 11
  - count 12
  - data 13
  - key 12
- Auxiliary Byte 32, 44

## B

- battery disposal ix
- Beginning of Extent Address 28
- Blocksize 26
- Bypass Cache 26

## C

- Cache Fast Write Data
  - Activate Cache Fast Write Access for Subsystem 110
  - Commit 113
  - Deactivate Cache Fast Write Access for Subsystem 110
  - Discard 114
  - identification 27
- Cache Fast Write Data ID
  - parameter 27
  - Read Subsystem Data 142
- cache operation mode
  - See subsystem operation mode
- caution notice ix
- CC 13
- changes, summary of xv
- Channel Initiated Command Retry
  - Overrun on Channel 173
- CKD conversion mode 25, 35, 36, 81, 82, 84, 85
- Coded ERP description 212
- command
  - descriptions 19—107
    - Define Extent 23—29
    - Device Release 100
    - Device Reserve 99
    - Diagnostic Control 90—95
    - Erase 80
    - Locate Record 29
    - Locate Record Extended 41—50
    - No-Operation 88
    - Perform Subsystem Function 111—134
    - Read and Reset Buffered Log 89
    - Read CKD 63
    - Read Configuration Data 95
    - Read Count 66
    - Read Data 65
    - Read Device Characteristics 54
    - Read Home Address 62
    - Read IPL 70
    - Read Key and Data 64
      - Read Message ID 153
    - Read Multiple CKD 67
    - Read Record Zero 62
    - Read Sector 54
    - Read Special Home Address 61
    - Read Subsystem Data 139—153
    - Read Track 68
    - Reset Allegiance 102
    - Search Home Address Equal 71
    - Search ID Equal 72
    - Search ID Equal or High 73

- command (*continued*)
  - descriptions (*continued*)
    - Search ID High 73
    - Search Key Equal 73
    - Search Key Equal or High 74
    - Search Key High 74
    - Seek 51
    - Seek Cylinder 51
    - Seek Head 52
    - Sense 86
    - Sense ID 87
    - Sense Path Group ID 105
    - Set File Mask 53
    - Set Path Group ID 104
    - Set Subsystem Mode 107—111
    - Suspend Multi-path Reconnection 106
    - Unconditional Reserve 101
    - Write CKD 77
    - Write CKD Next Track 79
    - Write Data 84
    - Write Home Address 76
    - Write Key and Data 82
    - Write Record Zero 76
    - Write Special Home Address 75
    - Write Update Data 83
    - Write Update Key and Data 81
  - mnemonics 19
  - summary list 19
- command information word 88
- command operations 22
- Commit 113
- communications statement 225
- compliance statement, radio frequency energy 225
- Compound Program Action Codes 213
- concurrent download of licensed internal code 10
- connect and disconnect functions, MPLF 155
- Connect order 128
- connection 4
- contingent allegiance 102
- Copy Services, ESS 6
- count
  - area 12
  - end-of-track before 81, 82, 84
  - Locate Record parameter 32
  - write commands 74, 77, 79
- cylinder 13

## D

- danger notice ix
- DASD Fast Write
  - Force Deactivate DASD Fast Write for Device 111
- data
  - access 14
  - area 13
  - length 12, 27

- data (*continued*)
  - length field = 0 78, 79, 84, 85, 86
  - transfer 16
  - write commands 74, 77, 79, 81, 82
- Deactivate Cache Fast Write Access for Subsystem 110
- Define Extent command
  - command description 23—29
  - overview 16
  - parameters 24—29
    - Beginning of Extent Address 28
    - Blocksize 26
    - End of Extent Address 29
    - Global Attributes Additional 27
    - Global Attributes Extended 27
    - Identification Bytes 27
    - Mask byte 24
- definition of serial number in sense data 192
- destage
  - Destage Modified Tracks order 118
  - processing errors 118
- Device Facilities
  - 3390-3 221
  - 3390-9 221
  - RAMAC 221
- Device Geometries
  - 3390-1 221
  - 3390-2 221
- Device Release command 100
- Device Reserve command 99
- device support
  - access authorization 24, 14, 29, 36
- diagnostic
  - access authorization 24, 61, 75, 92, 93, 94
  - track facility 24
- Diagnostic Control
  - OC 94
    - command description 90—95
    - Diagnostic Initialize Subsystem 93
    - Enable Write 93
    - Inhibit Write 92
    - Track Emulation Control 93
- Diagnostic Initialize Subsystem 93
- Discard 114
- discard, product ix
- disconnect and connect functions, MPLF 155
- Disconnect order 131
- DL
  - See data length
- documents, ordering xii
- domain 16

## E

- electronic emission notices 225



- emulation mode 54
- Enable Write 93
- enclosure, expansion 2
- End of Extent Address 29
- end-of-file 18
  - length field = 0 78, 79
  - record 18, 79
- end-of-track 14, 36, 38, 41, 63, 64, 66, 67, 68
- environmental notices ix
- erase 17
  - command 80
- Erase, Locate Record operation 39
- ESS Copy Services 6
- ESS Copy Services, accessing 5
- ESS Expert 7
- ESS product library xi
- ESS Specialist 6
- ESS Specialist, accessing 5
- ESS, host systems supported 3
- ESSNet (IBM Enterprise Storage Server Network) 4
- European Community compliance statement 226
- exception
  - conditions, identifying 22
  - conditions, priority of 22
  - outside Locate Record domain 23
  - within Locate Record domain 23
- expansion enclosure 2
- expansion enclosure, picture 2
- Expert, ESS 7
- extended operation byte 46
- Extended Parameter 46
- Extended Parameter Length 46
- extended-identification information 88
- extent 16
  - beginning of 28
  - end of 29

**F**

- failover 8
- Federal Communications Commission (FCC)
  - statement 225
- Fencing
  - OC subcommand 94
  - Storage Control Status 140
- fibre-channel host systems 3
- file
  - mask 17, 24, 53
  - protection 17
- Force Deactivate DASD Fast Write for Device 111
  - command description 111
- Format Write, Locate Record operation 36
- formatting 17
  - write commands 17, 74

## G

- Global Attributes Additional 27
- Global Attributes Extended 27

## H

- HA
  - See Home Address
- hex codes for commands 19
- HH 13
- Home Address 14, 13
  - orientation 31, 43
- host systems supported by the IBM ESS 3
- host systems, fibre-channel 3
- host systems, S/390 4
- host systems, SCSI 3

## I

- IBM Enterprise Storage Server Network (ESSNet) 4
- IBM Enterprise Storage Server overview 1
- IBM Subsystem Device Driver, overview 8
- Index 14
  - orientation 31, 43
- Industry Canada compliance statement 226
- Inhibit Cache Loading 26
- Inhibit Write 92
- interface, service 7
- interfaces, Web 4

## J

- Japanese Voluntary Control Council for Interference (VCCI) statement 227

## K

- key
  - area 12
  - length 12, 27
  - length field = 0 58, 78, 79
  - write commands 74, 77, 79, 81, 82
- KL
  - See key length
- Korean Government Ministry of Communication (MOC)
  - statement 227

## L

- licensed internal code, agreement 229
- list of ESS commands 19
- list of ESS publications xi
- Locate Record command
  - command definition 29
  - description 16
  - domain 16
    - Count parameter 32

Locate Record command (*continued*)  
 domain (*continued*)  
   execution within 33  
   Read Count suffix 32, 44  
 operations 34  
   Erase (11) 39  
   Format Write (03) 36  
   Orient (00) 35  
   Read (16) 40  
   Read Data (06) 38  
   Read Tracks (0C) 39  
   Write Data (01) 35  
   Write Track (0B) 38  
 parameters  
   Auxiliary Byte 32  
   Count 32  
   operation byte 30  
   Search Argument 33  
   Sector Number 33  
   Seek Address 33  
   Transfer Length Factor 33  
 search operation 34  
 Locate Record Extended command  
 command definition 41—50  
 description 16  
 LRE operations 47  
   Read Any 47  
   Read Trackset 50  
   Write Any 48  
 LRE parameters 42—46  
   Auxiliary Byte 44  
   extended operation byte 46  
   Extended Parameter 46  
   Extended Parameter Length 46  
   LRE count byte 44  
   operation byte 42  
   Search Argument 45  
   Sector Number 45  
   Seek Address 45  
   TLF 45  
 operations 50  
 lock state processing 156  
 Long Busy  
   See State change pending  
 LRE count byte 44  
 LRE Search Argument 45  
 LRE Sector Number 45  
 LRE Seek Address 45  
 LRE TLF 45

## M

manuals, list of xi  
 manuals, ordering xii  
 Mask byte 18, 24, 53

mnemonics, command 19  
 Mode, TPF 109  
 MPLF  
   See Multi-Path Lock Facility  
 Multi-Path Lock Facility  
   connect and disconnect functions 155  
   Connect order 128  
   creating and deleting Multi-Path Locks 156  
   Disconnect order 131  
   functional description 155  
   holder 166  
   lock command responses 159  
   Multi-Path Lock Facility support bit 56  
   Prepare to Read Lock Data order 120  
   Purge Lock order 125  
   Record Mode Control for TPF bit 109  
   Regular Data Format Mode 28  
   Set Lock State order 122  
   set lock state processing 156  
   state transformations 156  
   structure 155  
   terminology 165  
   TPF Mode 28  
   Unlock order 127  
   unlock processing 158  
   waiter 167  
 multi-track operations 22

## N

Native Mode 93  
 No-Operation command 88  
 Node Descriptor Record 151  
 non-volatile storage  
   Force Deactivate DASD Fast Write for Device 111  
 normal  
   access authorization 24  
   MPLF operation 27  
   Normal Cache Replacement 26  
 notice, caution ix  
 notice, danger ix  
 notices 223  
 notices, electronic emission 225  
 notices, environmental ix  
 notices, safety ix

## O

operation byte 30, 42  
 ordering publications xii  
 orders, PSF  
   See Perform Subsystem Function  
 Orient, Locate Record operation 35  
 orientation 15  
   count area 31, 43  
   data area 31, 43

orientation (*continued*)  
Home Address 31, 43  
index 31, 43  
modifiers 31, 43  
overview, IBM Enterprise Storage Server 1

## P

PAC  
See program actions  
parameters  
Define Extent 24—29  
Locate Record 30—34  
Locate Record Extended 42—46  
Partition Status Area  
Data Check or format error 188  
path recovery algorithms 10  
path selection policy 10  
PCI fetch mode 25  
Perform Subsystem Function  
Commit 113  
Connect 128  
Destage Modified Tracks 118  
Discard 114  
Disconnect 131  
Prepare for Read Subsystem Data 116  
Prepare to Read Lock Data 120  
Purge Lock 125  
Set Cache Allocation Parameters 119  
Set Lock State 122  
Set Special Intercept Condition 118  
Set-Interface-Identifier 133  
Suspend/Resume 120  
Unlock 127  
performance statistics  
Read Subsystem Data 141  
record description 141  
picture of the Models E10, E20, F10 and F20 front and rear views 2  
picture, expansion enclosure 2  
Prepare for Read Subsystem Data 116  
Prepare to Read Lock Data order (multi-path lock) 120  
primary track 14  
priority of exceptions 22  
product disposal ix  
product recycling ix  
program actions  
Compound Program Action Codes 213  
Single Program Action Codes 215  
PSF  
See Perform Subsystem Function  
publication list xi  
publications, ordering xii  
publications, related xii  
Purge Lock order 125

## R

R0  
See Record Zero  
radio-frequency energy compliance statement 225  
RAMAC  
device characteristics 219  
device facilities 221  
device geometry 221  
SNSID values 221  
Read and Reset Buffered Log command 89  
Read Any, LRE Operation 47  
Read CFWID 142  
Read CKD command 63  
read commands 60—70  
Read Configuration Data command 95  
Read Count  
command 66  
suffix 32, 44  
Read Data  
command 65  
Locate Record operation 38  
Read Device Characteristics  
characteristic values 219  
command 54  
Read Home Address command 62  
Read IPL command 70  
Read Key and Data command 64  
Read Logical Path Status 147  
Read Message Buffer 142  
Multi-Path Lock Facility Format 144  
Read Message ID 153  
Read MPLF Partition Data 150  
Read Multi-Path Lock Data suborder (multi-path lock) 149  
Read Multi-Path Lock User ID Data suborder (multi-path lock) 148  
Read Multiple CKD command 67  
Read Performance Statistics 141  
Read Pinned Track Data 147  
Read Record Zero command 62  
Read Sector command 54  
Read Special Home Address command 61  
Read Storage Control - Path Status 140  
Read Subsystem Data 139—153  
CFWID 142  
Logical Path Status 147  
Message Buffer 142  
Node Descriptor Record 151  
Performance Statistics 141  
Pinned Track Data 147  
Read MPLF Partition Data 150  
Read Multi-Path Lock Data 149  
Read Multi-Path Lock User ID Data 148  
Storage Control Path Status 140

- Read Track command 68
- Read Tracks, Locate Record operation 39
- Read Trackset, LRE Operation 50
- Read, Locate Record operation 40
- record 11
  - accessing 16
  - end-of-file 18
  - format 11
  - ID 12
  - user data 13
- Record Access 26
- Record Zero 14, 13
- Recovery Actions 205
- recycling ix
- related publications xii
- remote services support 7
- reset
  - See Selective Reset
  - See System Reset
- Reset Allegiance command 102
- Reset Notification 206
  - Set Path Group ID 104
- Retry, Command 25
- rotational position sensing 15

## S

- S/390 host systems 4
- safety notices ix
- SCSI host systems 3
- search
  - argument, Locate Record 33
  - commands 70—74
  - operation 16
- Search Home Address Equal command 71
- Search ID Equal command 72
- Search ID Equal or High command 73
- Search ID High command 73
- Search Key Equal command 73
- Search Key Equal or High command 74
- Search Key High command 74
- sector 15
  - number, Locate Record parameter 33
- security, Web connection 4
- seek
  - address
    - Locate Record parameter 33
  - commands 50—52
- Seek command 51
- Seek Cylinder command 51
- Seek Head command 52
- Selective Reset
- Sense command 86
- Sense ID
  - characteristic values 221
  - command 87

- Sense Path Group ID command 105
- Sense Subsystem Status 134
- Sequential Access 26
  - Sequential Prestage 26
- serial number 192
- service interface 7
- services support, Remote 7
- Set Cache Allocation Parameters order 119
- Set File Mask command 53
- Set Lock State order 122
- Set Path Group ID command 104
- Set Special Intercept Condition 118
- Set Subsystem Mode
  - Activate Cache Fast Write Access for Subsystem 110
    - command description 107
  - Deactivate Cache Fast Write Access for Subsystem 110
    - Force Deactivate DASD Fast Write for Device 111
- Set-Interface-Identifier 133
- Single Program Action Codes 215
- Specialist, ESS 6
- State change pending
  - format F, message 0 sense 185
  - State change pending Action code 1D 217
- state transformations, MPLF 156
- statement of compliance 225, 226, 227
  - Canadian compliance statement 226
  - Federal Communications Commission 225
  - Korean Government Ministry of Communication (MOC) 227
- statement of compliance, Japan 227
- status
  - unit exception 18
- subsystem operation mode
  - Bypass Cache 26
  - Inhibit Cache Loading 26
  - Normal Cache Replacement 26
  - Record Access 26
  - Sequential Access 26
  - Sequential Prestage 26
- suffix, Read Count 32, 44
- summary of changes xv
- summary of ESS commands 19
- support, remote services 7
- Suspend Multi-path Reconnection command 106
- Suspend/Resume order 120
- System Reset
- systems, S/390 host 4

## T

- TPF Mode 28, 56, 109, 119, 120, 122, 125, 127, 128, 131, 135, 141, 155
- track
  - accessing 14

- track (*continued*)
  - address 13
  - definition 11
  - end-of-track 14
  - format 13
  - format exception
    - condition 02 81, 82, 84, 85
    - condition 03 36
  - group 13
  - index 14
  - multi-track operations 22
  - primary 14
  - status
    - Data Check or format error 188
    - Track Emulation Control 93
- trademarks 225
- Transfer Length Factor 33

## U

- Unconditional Reserve command 101
- Unfence 94
- unit exception 18
- Unlock order 127
- update write commands 74
- user data record 13

## V

- VPD 119, 120, 122, 125, 127, 128, 131, 155

## W

- Web connection security 4
- Web interfaces 4
- write
  - commands
    - descriptions 74—86
    - Write CKD Next Track 79
    - Write Count, Key, and Data 77
    - Write Data 84
    - Write Home Address 76
    - Write Key and Data 82
    - Write Record Zero 76
    - Write Special Home Address 75
    - Write Update Data 83
    - Write Update Key and Data 81
  - operations
    - Write Any, Locate Record Extended 48
    - Write Data, Locate Record 35
    - Write Track, Locate Record 38

---

# Communicating Your Comments to IBM

IBM Enterprise Storage Server™  
System/390 Command Reference  
2105 Models E10, E20, F10, and F20  
Publication No. SC26-7298-01

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 this 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 at the back of this book.
- If you prefer to send comments by FAX, use this number:
  - United States: 1-800-426-6209
  - Other countries: (+1)+408+256-7896
- If you prefer to send comments electronically, use this network ID:
  - IBMLink from Canada: STARPUBS at TORIBM
  - IBM Mail Exchange: USIB3VVD at IBMMAIL
  - Internet: starpubs@us.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 Enterprise Storage Server™  
System/390 Command Reference  
2105 Models E10, E20, F10, and F20  
Publication No. SC26-7298-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

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.



Cut or Fold  
Along Line

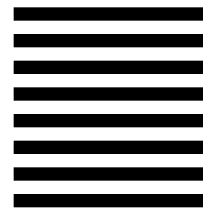
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES



# BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation  
RCF Processing Department  
G26/050  
5600 Cottle Road  
SAN JOSE, CA 95193-0001



Fold and Tape

Please do not staple

Fold and Tape

Cut or Fold  
Along Line







Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

SC26-7298-01

