
Date: 22 July 1999
To: T10 Technical Committee
From: Ralph Weber, LSI Logic Alternate Member of T10
Subj: EXTENDED COPY command for SPC-2

This revision contains those changes agreed by the July SCSI Working Group. The changes are the addition of definitions for world wide name and N_Port. As a result of correcting the definition of N_Port, further changes were necessary for clause and table titles. Also, it was agreed that N_Port is the only correct spelling, i.e., uses of N_port have been changed to N_Port.

I also undertook one last review of the entire proposal during which several editorial improvements were made.

This page left blank intentionally

## 7.1 EXTENDED COPY command

The EXTENDED COPY command (see table 8) provides a means to copy data from one set of logical units to another set or to the same set of logical units. The entity within a device server that receives and performs the EXTENDED COPY command is called the copy manager. The copy manager is responsible for copying data from the source device(s) to the destination device(s). The copy source and destination devices are logical units that may reside in different SCSI devices or the same SCSI device (in fact all the devices and the copy manager may be the same logical unit).

**Table 8 — EXTENTED COPY command**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | OPERATION CODE (83h) | | | | | | | |
| 1 | Reserved | | | | | | | |
| 2 | Reserved | | | | | | | |
| 3 | Reserved | | | | | | | |
| 4 | Reserved | | | | | | | |
| 5 | Reserved | | | | | | | |
| 6 | Reserved | | | | | | | |
| 7 | Reserved | | | | | | | |
| 8 | Reserved | | | | | | | |
| 9 | Reserved | | | | | | | |
| 10 | (MSB) | | | PARAMETER LIST LENGTH | | | | |
| 11 | | | | | | | | |
| 12 | | | | | | | | |
| 13 | | | | | | | | (LSB) |
| 14 | Reserved | | | | | | | |
| 15 | CONTROL | | | | | | | |

Before the copy manager is instructed to move data, the application controlling the data movement shall independently execute any activities necessary to prepare the source and destination devices for the EXTENDED COPY command. These activities could include media changer commands, loading of tapes, MODE SELECT commands reservation commands, positioning of tape, and etc. After all preparatory actions have been accomplished, the EXTENDED COPY command should be issued to the copy manager to start the data transfer.

The PARAMETER LIST LENGTH field specifies the length in bytes of the parameter data that shall be contained in the Data-Out Buffer. A parameter list length of zero indicates that no data shall be transferred. This condition shall not be considered as an error.

The EXTENDED COPY parameter list (see table 9) begins with a twelve byte header that contains the LIST IDENTIFIER field, the STR, and NRCR bits, the command's priority, the length of the target descriptor list, the length of

the segment descriptor list, and the length of the optional inline data. Immediately following the header is one or more target descriptors, followed by one or more segment descriptors, followed by any optional inline data.

**Table 9 — EXTENDED COPY parameter list**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | LIST IDENTIFIER | | | | | | | |
| 1 | Reserved | | STR | NRCR | Reserved | PRIORITY | | |
| 2 | (MSB) | | | TARGET DESCRIPTOR LIST LENGTH (n-12) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | | | SEGMENT DESCRIPTOR LIST LENGTH (m-k+1) | | | | |
| 5 | | | | | | | | |
| 6 | | | | | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | (MSB) | | | INLINE DATA LENGTH | | | | |
| 9 | | | | | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |
| | Target descriptor(s) | | | | | | | |
| 12 | Target descriptor 0 | | | | | | | |
| 43 | | | | | | | | |
| | . . | | | | | | | |
| n-31 | Target descriptor x | | | | | | | |
| n | | | | | | | | |
| | Segment descriptor(s) | | | | | | | |
| k | Segment descriptor 0 (See specific table for length.) | | | | | | | |
| k+l | | | | | | | | |
| | . . | | | | | | | |
| m | Segment descriptor y (See specific table for length.) | | | | | | | |
| | Inline data | | | | | | | |

The LIST IDENTIFIER field is a unique value selected by the application client to identify the extended copy operation to the copy manager. The list identifier also may be used in the RECEIVE COPY RESULTS command (see 7.77) to request status for a specific EXTENDED COPY command. The LIST IDENTIFIER value shall be unique for each concurrent EXTENDED COPY command sent by an initiator. If the copy manager detects a duplicate LIST IDENTIFIER value the command shall be terminated with a CHECK CONDITION, the sense key shall be set to ILLEGAL REQUEST and the additional sense data shall be set to OPERATION IN PROGRESS.

The PRIORITY field establishes the priority of this EXTENDED COPY command relative to other commands being executed by the same device server. All commands other than copy commands have a priority of 1. Priority 0 is the highest priority, with increasing PRIORITY values indicating lower priorities.

A Sequential Stripped bit (STR) value of one indicates to the copy manager that most of the disk references in the parameter list represent sequential access of several stripped disks. This may be used by the copy manager to implement read-ahead. A STR value of zero indicates to the copy manager that disk references are not necessarily sequential.

If the No Receive Copy Results (NRCR) bit is one, the application client shall not send a RECEIVE COPY RESULTS command to collect the results of this parameter list. If NRCR is zero, the application client may send RECEIVE COPY RESULTS command to receive the results of this parameter list.

The TARGET DESCRIPTOR LIST LENGTH contains the length in bytes of the target descriptor list that immediately follows the parameter list header. The number of target descriptors equals the length in bytes of the target descriptor list divided by 32.

An EXTENDED COPY command may reference one or more target devices (which are the source and/or the destination logical units). Each target device is described by a target descriptor. All target descriptors have their formats specified by an EXTENDED COPY descriptor code. A copy manager need not support all target descriptor formats. See 7.1.4 for a detailed description of the target descriptors.

Segment descriptors reference target descriptors by their position, or index, in the target descriptor list. The index for a target descriptor is the starting byte number for the target descriptor in the parameter data minus 12 divided by 32. The maximum number of target descriptors permitted within a parameter list is indicated by the MAXIMUM TARGET COUNT field in the copy manager's operating parameters (see 7.77.3). If the number of target descriptors exceeds the allowed number, the command shall be terminated with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to TOO MANY TARGET DESCRIPTORS (26h/06h).

The SEGMENT DESCRIPTOR LIST LENGTH contains the length in bytes of the segment descriptor list that follows the target descriptors. See 7.1.5 for a detailed description of the segment descriptors. The maximum number of segment descriptors permitted within a parameter list is indicated by the MAXIMUM SEGMENT COUNT field in the copy manager's operating parameters (see 7.77.3). If the number of segment descriptors exceeds the allowed number, the command shall be terminated with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to TOO MANY SEGMENT DESCRIPTORS (26h/08h).

The maximum length of the descriptors (both target and segment) permitted within a parameter list is indicated by the MAXIMUM DESCRIPTOR LIST LENGTH field in the copy manager's operating parameters (see 7.77.3). If the combined length of the target and segment descriptors exceeds the allowed value, the command shall be terminated with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to PARAMETER LIST LENGTH ERROR.

The INLINE DATA LENGTH field contains the number of bytes of Inline data, after the last segment descriptor. A value of zero indicates that no inline data is present.

The copy manager shall move data from the source devices to the destination devices in the manner proscribed by the segment descriptors. The specific commands issued by the copy manager to the source and destination devices while processing the segment descriptors is vendor specific. Upon completion of an EXTENDED COPY command that returns GOOD status, the source and destination devices (particularly stream devices) shall be positioned at deterministic locations such that the device could be repositioned to the same location by the application client with appropriate commands.

### 7.1.1 Errors detected before starting processing of the segment descriptors

Errors may occur during processing of an EXTENDED COPY command before the first segment descriptor is processed. These conditions include parity errors while transferring the EXTENDED COPY command, invalid parameters in the CDB or parameter data, invalid segment descriptors, and inability of the copy manager to continue operating. In the event of such an exception condition, the copy manager shall:

a) terminate the EXTENDED COPY command with CHECK CONDITION status; and
b) set the VALID bit in the sense data to zero. The sense key shall contain the sense key code describing the exception condition (i.e.: not COPY ABORTED).

### 7.1.2 Errors detected during processing of segment descriptors

Errors may occur after the copy manager has begun processing segment descriptors. These include invalid parameters in segment descriptors, invalid segment descriptors, unavailable targets referenced by target descriptors, inability of the copy manager to continue operating, and errors reported by source or destination target devices. If the copy manager receives a CHECK CONDITION status from one of the target devices, it shall recover the sense data associated with the exception condition and clear the ACA condition (if any) associated with the CHECK CONDITION status.

Following an exception condition detected during segment descriptor processing, the copy manager shall:

a) terminate the EXTENDED COPY command with CHECK CONDITION status;
b) set the sense key code to COPY ABORTED;
c) indicate the segment that was being processed at the time of the exception by writing the segment number to third and forth bytes of the COMMAND-SPECIFIC INFORMATION field. The segment number is based on the relative position of the segment descriptor in the EXTENDED COPY parameter list. The first segment descriptor in the parameter list is assigned descriptor number zero, the second is assigned one, etc.;
d) If any data has been transferred for the segment being processed at the time the error occurred, the residual for the segment shall be placed in the INFORMATION field, and the VALID bit shall be set to 1. If the segment descriptor specifies a transfer count in blocks, then the residual count is the number of blocks remaining for transfer, otherwise, the residual count is the number of bytes remaining for transfer. If no data has been transferred for the segment being processed at the time the error occurred, then the VALID bit shall be set to 0 and the contents of the INFORMATION field are not defined. Segment descriptors that do not specify a transfer count shall not have a valid residual count returned;
e) If the exception condition is reported by the source logical unit, then the first byte of the COMMAND-SPECIFIC INFORMATION field shall specify the starting byte number, relative to the first byte of sense data, of an area that contains (unchanged) the source logical unit's status byte and sense data. A zero value indicates that no status byte and sense data is being returned for the source logical unit;
f) If the exception condition is reported by the destination logical unit, then the second byte of the COMMAND-SPECIFIC INFORMATION field shall specify the starting byte number, relative to the first byte of sense data, of an area that contains (unchanged) the destination logical unit's status byte and sense data. A zero value indicates that no status byte and sense data is being returned for the destination logical unit;
g) If, during the processing of a segment descriptor, the copy manager determines that a target is not reachable, then the SENSE-KEY SPECIFIC field shall be set as described in 7.20.1, with the FIELD POINTER field indicating the first byte of the target descriptor that identifies the target; and
h) If, during the processing of a segment descriptor, the copy manager detects an error in the segment descriptor, then the SENSE-KEY SPECIFIC field shall be set as described in 7.20.1, with the FIELD POINTER field indicating byte in error. The FIELD POINTER field may be used to indicate an offset into either the parameter data or the segment descriptor. The SD bit is used to differentiate between these two cases. The SD bit shall be set to zero to indicate the FIELD POINTER field contains an offset from the start of the parameter data. The SD bit shall be set to one to indicate the FIELD POINTER field contains an offset from the start of the segment descriptor.

### 7.1.3 Descriptor type codes

Target descriptors and segment descriptors share a single set of code values that identify the type of descriptor (see table 10). Segment descriptors use codes in the range 00h to BFh. The definitions of codes between C0h and DFh are vendor specific. Target descriptors use codes in the range E0h to FFh.

**Table 10 — EXTENDED COPY descriptor type codes**

| Descriptor type code | Reference | Description [1] | Shorthand [1] |
|---|---|---|---|
| 00h | 7.1.5.1 | Copy from block device to stream device | block→stream |
| 01h | 7.1.5.2 | Copy from stream device to block device | stream→block |
| 02h | 7.1.5.3 | Copy from block device to block device | block→block |
| 03h | 7.1.5.4 | Copy from stream device to stream device | stream→stream |
| 04h | 7.1.5.5 | Copy inline data to stream device | inline→stream |
| 05h | 7.1.5.6 | Copy embedded data to stream device | embedded→stream |
| 06h | 7.1.5.7 | Read from stream device and discard | stream→discard |
| 07h | 7.1.5.8 | Verify block or stream device operation | |
| 08h | 7.1.5.9 | Copy block device with offset to stream device | block<o>→stream |
| 09h | 7.1.5.10 | Copy stream device to block device with offset | stream→block<o> |
| 0Ah | 7.1.5.11 | Copy block device with offset to block device with offset | block<o>→block<o> |
| 0Bh | 7.1.5.1 | Copy from block device to stream device and hold a copy of read data for the application client **[2]** | block→stream +application client |
| 0Ch | 7.1.5.2 | Copy from stream device to block device and hold a copy of read data for the application client **[2]** | stream→block +application client |
| 0Dh | 7.1.5.3 | Copy from block device to block device and hold a copy of read data for the application client **[2]** | block→block +application client |
| 0Eh | 7.1.5.4 | Copy from stream device to stream device and hold a copy of read data for the application client **[2]** | stream→stream +application client |
| 0Fh | 7.1.5.7 | Read from stream device and hold a copy of read data for the application client **[2]** | stream→discard +application client |
| 10h | 7.1.5.12 | Write filemarks to sequential-access device | filemark→tape |
| 11h | 7.1.5.13 | Space records or filemarks on sequential-access device | space→tape |
| 12h | 7.1.5.14 | Locate on sequential-access device | locate→tape |

NOTES

**[1]** Block devices are those with peripheral device type codes 0h, 4h, 5h, 7h, and Eh. Stream devices are those devices with peripheral device type codes 1h and 3h. Sequential-access (tape) devices are those with peripheral device type code 01h. See 7.4.1 for peripheral device type code definitions.

**[2]** The application client shall use the RECEIVE COPY RESULTS with a RECEIVE DATA service action to retrieve data held for it by the copy manager (see 7.77.2).

**Table 10 — EXTENDED COPY descriptor type codes**

| Descriptor type code | Reference | Description [1] | Shorthand [1] |
|---|---|---|---|
| 13h | 7.1.5.15 | Image copy from sequential-access device to sequential-access device | <i>tape→<i>tape |
| 14h - BFh | | Reserved for segment descriptors | |
| C0h - DFh | | Vendor unique descriptors | |
| E0h | 7.1.4.1 | World Wide Name target descriptor | |
| E1h | 7.1.4.2 | N_Port target descriptor | |
| E2h | 7.1.4.3 | N_Port with World Wide Name checking target descriptor | |
| E3h | 7.1.4.4 | Parallel Interface T_L target descriptor | |
| E4h | 7.1.4.5 | Identification descriptor target descriptor | |
| E5h - FFh | | Reserved for target descriptors | |

NOTES
**[1]** Block devices are those with peripheral device type codes 0h, 4h, 5h, 7h, and Eh. Stream devices are those devices with peripheral device type codes 1h and 3h. Sequential-access (tape) devices are those with peripheral device type code 01h. See 7.4.1 for peripheral device type code definitions.
**[2]** The application client shall use the RECEIVE COPY RESULTS with a RECEIVE DATA service action to retrieve data held for it by the copy manager (see 7.77.2).

### 7.1.4 Target descriptors

All target descriptors are 32 bytes in length and begin with a four-byte header (see table 11) that contains the DESCRIPTOR TYPE CODE field, which identifies the format of the descriptor. The assigned values for target descriptors type codes are shown in table 10. Support for each target descriptor format is optional. If copy manager receives an unsupported descriptor type code in a target descriptor, the command shall be terminated with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to UNSUPPORTED TARGET DESCRIPTOR TYPE CODE (26h/07h).

**Table 11 — Target descriptor format**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (E0 - FFh) | | | | | | | |
| 1 | Reserved | | | PERIPHERAL DEVICE TYPE | | | | |
| 2 | Reserved | | | | | | | |
| 3 | Reserved | | | | | | | |
| 4 | Target descriptor parameters | | | | | | | |
| 27 | | | | | | | | |
| 28 | Device type specific parameters | | | | | | | |
| 31 | | | | | | | | |

The DESCRIPTOR TYPE CODE field is described in 7.1.3. The PERIPHERAL DEVICE TYPE field is described in 7.4.1. The value in the DESCRIPTOR TYPE CODE field determines the format of the target descriptor parameters that follow the four-byte header and precede the device type specific parameters. The values in the DESCRIPTOR TYPE CODE field are listed in table 10.

The value in the PERIPHERAL DEVICE TYPE field determines the format of the device type specific parameters that follow the target descriptor parameters. The device type specific parameters convey information specific to the type of device identified by the target descriptor. Table 12 lists the peripheral device type code values having formats defined for the device type specific parameters in a target descriptor. Peripheral device types with code values not listed in table 12 are reserved in the EXTENDED COPY parameter list.

**Table 12 — Device type specific parameters in target descriptors**

| Peripheral Device Type | Reference | Description | Shorthand |
|---|---|---|---|
| 00h, 04h, 05h, 07h, and 0Eh | 7.1.4.6 | Block devices | Block |
| 01h | 7.1.4.7 | Sequential-access devices | Stream or Tape |
| 03h | 7.1.4.8 | Processor devices | Stream |

The copy manager may, prior to processing a segment descriptor, verify the information in a target descriptor's device specific fields. However, the copy manager shall not issue any commands that change the state of the target device to verify the information.

### 7.1.4.1 World Wide Name target descriptor format

The target descriptor format shown in table 13 is used to identify a target using its World Wide Name.

**Table 13 — World Wide Name target descriptor format**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (E0h) | | | | | | | |
| 1 | Reserved | | | PERIPHERAL DEVICE TYPE | | | | |
| 2 | Reserved | | | | | | | |
| 3 | Reserved | | | | | | | |
| 4 | (MSB) | | | LOGICAL UNIT NUMBER | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | (MSB) | | | WORLD WIDE NAME | | | | |
| 19 | | | | | | | | (LSB) |
| 20 | | | | Reserved | | | | |
| 27 | | | | | | | | |
| 28 | | | | Device type specific parameters | | | | |
| 31 | | | | | | | | |

The DESCRIPTOR TYPE CODE and PERIPHERAL DEVICE TYPE fields and the device type specific parameters are described in 7.1.4.

The LOGICAL UNIT NUMBER field specifies the logical unit within the SCSI device addressed by the data in the WORLD WIDE NAME field that shall be the target (source or destination) for EXTENDED COPY operations.

The WORLD WIDE NAME field shall contain the port World Wide Name defined by the Physical Log In (PLOGI) extended link service, defined in FC-PH.

NOTE 4 The World Wide Name target descriptor format burdens the copy manager with translating the World Wide Name to an N_Port identifier (see 7.1.4.2).

### 7.1.4.2 N_Port target descriptor format

The target descriptor format shown in table 14 is used to identify a target using its N_Port.

**Table 14 — N_Port target descriptor format**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (E1h) | | | | | | | |
| 1 | Reserved | | | PERIPHERAL DEVICE TYPE | | | | |
| 2 | Reserved | | | | | | | |
| 3 | Reserved | | | | | | | |
| 4 | (MSB) | | | LOGICAL UNIT NUMBER | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | | | | Reserved | | | | |
| 20 | | | | | | | | |
| 21 | (MSB) | | | | | | | |
| 22 | | | | N_PORT | | | | |
| 23 | | | | | | | | (LSB) |
| 24 | | | | Reserved | | | | |
| 27 | | | | | | | | |
| 28 | | | | Device type specific parameters | | | | |
| 31 | | | | | | | | |

The DESCRIPTOR TYPE CODE and PERIPHERAL DEVICE TYPE fields and the device type specific parameters are described in 7.1.4.

The LOGICAL UNIT NUMBER field specifies the logical unit within the SCSI device addressed by the data in the N_PORT field that shall be the target (source or destination) for EXTENDED COPY operations.

The N_PORT field shall contain the FC-PH port D_ID to be used to transport frames including PLOGI and FCP-2 related frames.

### 7.1.4.3 N_Port with World Wide Name checking target descriptor format

Targets addressed using their N_Port with World Wide Name checking are identified using the target descriptor format shown in table 15.

**Table 15 — N_Port with World Wide Name checking target descriptor format**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (E2h) | | | | | | | |
| 1 | Reserved | | | PERIPHERAL DEVICE TYPE | | | | |
| 2 | Reserved | | | | | | | |
| 3 | Reserved | | | | | | | |
| 4 | (MSB) | | | LOGICAL UNIT NUMBER | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | (MSB) | | | WORLD WIDE NAME | | | | |
| 19 | | | | | | | | (LSB) |
| 20 | Reserved | | | | | | | |
| 21 | (MSB) | | | N_PORT | | | | |
| 22 | | | | | | | | |
| 23 | | | | | | | | (LSB) |
| 24 | Reserved | | | | | | | |
| 27 | | | | | | | | |
| 28 | Device type specific parameters | | | | | | | |
| 31 | | | | | | | | |

The DESCRIPTOR TYPE CODE and PERIPHERAL DEVICE TYPE fields and the device type specific parameters are described in 7.1.4.

The LOGICAL UNIT NUMBER field specifies the logical unit with in the SCSI device addressed by the data in the N_PORT and WORLD WIDE NAME and fields that shall be the target (source or destination) for EXTENDED COPY operations.

The WORLD WIDE NAME field shall contain the port World Wide Name defined by the Physical Log In (PLOGI) extended link service, defined in FC-PH.

The N_PORT field shall contain the FC-PH port D_ID to be used to transport frames including PLOGI and FCP-2 related frames.

When the copy manager first processes a segment descriptor that references this target descriptor, it shall confirm that the D_ID in the N_PORT field is associated with the World Wide Name in the WORLD WIDE NAME field. If the association cannot be confirmed, the EXTENDED COPY command shall be terminated because the target is unavailable (see 7.1.2). The copy manager shall track configuration changes that affect the D_ID value for the duration of the EXTENDED COPY commands. An application client generating the EXTENDED COPY commands is responsible for tracking configuration changes between commands.

#### 7.1.4.4 Parallel Interface T_L target descriptor format

Targets addressed using their parallel SCSI bus Target ID, and logical unit number are identified using the target descriptor format shown in table 16.

**Table 16 — Parallel Interface T_L target descriptor format**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (E3h) | | | | | | | |
| 1 | Reserved | | | PERIPHERAL DEVICE TYPE | | | | |
| 2 | Reserved | | | | | | | |
| 3 | Reserved | | | | | | | |
| 4 | (MSB) | | | LOGICAL UNIT NUMBER | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | Vendor unique | | | | | | | |
| 13 | TARGET IDENTIFIER | | | | | | | |
| 14 | | | | Reserved | | | | |
| 27 | | | | | | | | |
| 28 | | | | Device type specific parameters | | | | |
| 31 | | | | | | | | |

The DESCRIPTOR TYPE CODE and PERIPHERAL DEVICE TYPE fields and the device type specific parameters are described in 7.1.4.

The LOGICAL UNIT NUMBER field specifies the logical unit with in the SCSI device addressed by the data in the TARGET IDENTIFIER field that shall be the target (source or destination) for EXTENDED COPY operations.

The TARGET IDENTIFIER field specifies the SCSI target identifier to be used when this target descriptor identifies the source or destination of an EXTENDED COPY operation.

### 7.1.4.5 Identification descriptor target descriptor format

The target descriptor format shown in table 17 instructs the copy manager to locate a target and logical unit that returns a device identification VPD page (see 8.4.3) containing an Identification descriptor having the specified CODE SET, ASSOCIATION, IDENTIFIER TYPE, IDENTIFIER LENGTH, and IDENTIFIER field values. The copy manager may use any N_port, target identifier and logical unit number values that result in matching VPD field values to address the copy device. If multiple N_port, target identifiers and logical unit number combinations access matching VPD field values, the copy manager may use any combination to address the copy device and shall try other combinations in the event that one combination becomes non-operational during the processing of an EXTENDED COPY command.

**Table 17 — Identification descriptor target descriptor format**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (E4h) | | | | | | | |
| 1 | Reserved | | | PERIPHERAL DEVICE TYPE | | | | |
| 2 | Reserved | | | | | | | |
| 3 | Reserved | | | | | | | |
| 4 | Reserved | | | | CODE SET | | | |
| 5 | Reserved | | ASSOCIATION | | IDENTIFIER TYPE | | | |
| 6 | Reserved | | | | | | | |
| 7 | IDENTIFIER LENGTH (n-7) | | | | | | | |
| 8 | (MSB) IDENTIFIER | | | | | | | |
| n | | | | | | | | (LSB) |
| n+1 | Reserved | | | | | | | |
| 27 | | | | | | | | |
| 28 | Device type specific parameters | | | | | | | |
| 31 | | | | | | | | |

The DESCRIPTOR TYPE CODE and PERIPHERAL DEVICE TYPE fields and the device type specific parameters are described in 7.1.4.

The contents of the CODE SET, ASSOCIATION, IDENTIFIER TYPE, IDENTIFIER LENGTH, and IDENTIFIER fields is specified in 8.4.3.

The identifier length shall be 20 or less. If the identifier length is 20 there shall be no reserved bytes between the target descriptor parameters and the device type specific parameters.

Some combinations of code set, association, identifier type, identifier length and identifier do not uniquely identify a logical unit to serve as a copy target device. The application client shall not send such combinations to the copy manager.

### 7.1.4.6 Device type specific target descriptor parameters for block device types

The format for the device type specific target descriptor parameters for block device types (device type code values 00h, 04h, 05h, 07h, and 0Eh) is shown in table 18.

**Table 18 — Device type specific target descriptor parameters for block device types**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 28 | Reserved | | | | | PAD | Reserved | |
| 29 | (MSB) | | | | | | | |
| 30 | DISK BLOCK LENGTH | | | | | | | |
| 31 | | | | | | | | (LSB) |

The PAD bit is used in conjunction with the CAT bit (see 7.1.5) in the segment descriptor to determine what action should be taken when a segment of the copy does not fit exactly into an integer number of destination blocks.

The DISK BLOCK LENGTH is the number of bytes in a disk block for the logical device being addressed.

### 7.1.4.7 Device type specific target descriptor parameters for stream device types

The format for the device type specific target descriptor parameters for the sequential-access device type (device type code value 01h) is shown in table 19.

**Table 19 — Device type specific target descriptor parameters for processor device types**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 28 | Reserved | | | | | PAD | SILI | FIXED |
| 29 | (MSB) | | | | | | | |
| 30 | STREAM BLOCK LENGTH | | | | | | | |
| 31 | | | | | | | | (LSB) |

The contents of the FIXED bit and STREAM BLOCK LENGTH field are combined with the STREAM DEVICE TRANSFER LENGTH field in the segment descriptor to determine the length of the stream read or write operation as specified in table 20.

**Table 20 — Stream device transfer lengths** (part 1 of 2)

| FIXED bit | STREAM BLOCK LENGTH field | Description |
|---|---|---|
| 0 | 0 | Use variable length reads or writes. The number bytes for each read or write is specified by the STREAM DEVICE TRANSFER LENGTH field in the segment descriptor. |
| 0 | not 0 | The command shall be terminated with a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST. |

**Table 20 — Stream device transfer lengths** (part 2 of 2)

| FIXED bit | STREAM BLOCK LENGTH field | Description |
|---|---|---|
| 1 | 0 | The command shall be terminated with a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN PARAMETER LIST |
| 1 | not 0 | Use fixed record length reads or writes. The number of bytes for each read or write shall be the product of the STREAM BLOCK LENGTH field and the STREAM DEVICE TRANSFER LENGTH field in the segment descriptor. |

The PAD bit is used in conjunction with the CAT bit (see 7.1.5) in the segment descriptor to determine what action should be taken when a segment of the copy does not fit exactly into an integer number of destination blocks.

The SILI bit indicates the value used in the SILI bit of any read commands issued to the target.

### 7.1.4.8 Device type specific target descriptor parameters for processor device types

The format for the device type specific target descriptor parameters for the processor device type (device type code value 03h) is shown in table 21.

**Table 21 — Device type specific target descriptor parameters for stream device types**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 28 | Reserved | | | | | PAD | Reserved | |
| 29 | Reserved | | | | | | | |
| 31 | | | | | | | | |

The PAD bit is used in conjunction with the CAT bit (see 7.1.5) in the segment descriptor to determine what action should be taken when a segment of the copy does not fit exactly into an integer number of SEND or RECEIVE commands.

When the processor device is a source, the number of bytes to be transferred by a SEND command shall be specified by STREAM DEVICE TRANSFER LENGTH field in the segment descriptor. When the processor device is a destination, the number of bytes to be transferred by a RECEIVE command shall be specified by STREAM DEVICE TRANSFER LENGTH field in the segment descriptor.

### 7.1.5 Segment Descriptors

All segment descriptors begin with the eight byte header shown in table 22.

**Table 22 — Segment descriptor header**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (00h-3Fh) | | | | | | | |
| 1 | Reserved | | | | | | DC | CAT |
| 2 | (MSB) | | | DESCRIPTOR LENGTH | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | | | SOURCE TARGET DESCRIPTOR INDEX | | | | |
| 5 | | | | | | | | (LSB) |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3. Support for each segment descriptor format is optional. If copy manager receives an unsupported descriptor type code in a segment descriptor, the command shall be terminated with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to UNSUPPORTED SEGMENT DESCRIPTOR TYPE CODE (26h/09h).

The destination count (DC) bit is used in those segment descriptors where both the source and destination devices have the same storage structure (block or stream). The DC bit is only applicable to segment descriptors with DESCRIPTOR TYPE CODE values of 02h, 03h, 0Dh, and 0Eh. The DC bit is reserved for all other segment descriptors. Details of usage for the DC bit appear in the clauses defining the segment descriptors that use it.

The CAT bit is used in conjunction with the PAD bit in the target descriptors to define what action should be taken when a segment of the copy does not fit exactly into an integral number of destination blocks. Table 23 defines the operation of the PAD and CAT bits.

**Table 23 — PAD and CAT bit definitions** (part 1 of 2)

| PAD bit in | | CAT bit | Copy manager action |
|---|---|---|---|
| Source target descriptor | Destination target descriptor | | |
| 1 | 0 | 0 | On inexact segments, the copy manager shall strip input characters from the final source block(s), always stopping at the end of a complete block. |
| 0 or 1 | 1 | 0 | On inexact segments, the copy manager shall add pad characters (00h) to the destination block to completely fill the block. |

**Table 23 — PAD and CAT bit definitions** (part 2 of 2)

| PAD bit in | | CAT bit | Copy manager action |
|---|---|---|---|
| Source target descriptor | Destination target descriptor | | |
| 0 or 1 | 0 or 1 | 1 | The copy manager shall always write or read complete blocks. On inexact segments, the remainder of the destination block data shall be taken from the next segment. If the CAT bit is one on the last segment of an EXTENDED COPY command the command shall be terminated with a CHECK CONDITION status. The sense key shall be set to an COPY ABORTED and the additional sense code shall be set to UNEXPECTED INEXACT SEGMENT (26h/0Ah). |
| 0 | 0 | 0 | The occurrence of an inexact segment when both PAD bits and the CAT bit are all zero shall be an error. The the command shall be terminated with a CHECK CONDITION status. The sense key shall be set to an COPY ABORTED and the additional sense code shall be set to UNEXPECTED INEXACT SEGMENT (26h/0Ah). |

The DESCRIPTOR LENGTH field contains the length in bytes of the fields that follow the DESCRIPTOR LENGTH field in the segment descriptor. In most cases, the length is constant.

The SOURCE TARGET DESCRIPTOR INDEX field contains an index into the target descriptor list (see 7.1) identifying the source target device. The DESTINATION TARGET DESCRIPTOR INDEX field contains an index into the target descriptor list (see 7.1) identifying the destination target device. Some segment descriptor formats do not require a SOURCE TARGET DESCRIPTOR INDEX field or a DESTINATION TARGET DESCRIPTOR INDEX field, in which case the field is reserved.

If the target identified by a SOURCE TARGET DESCRIPTOR INDEX field or a DESTINATION TARGET DESCRIPTOR INDEX field is not accessible to the copy manager, then the command shall be terminated with a CHECK CONDITION status, the sense key shall be set to COPY ABORTED and the additional sense code shall be set to UNREACHABLE COPY TARGET (08h/04h).

#### 7.1.5.1 Block device to stream device operations

The segment descriptor format shown in table 24 is used by the copy operations that move data from a block device to a stream device or vice versa.

**Table 24 — Block device to or from stream device segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (00h, 01h, 0Bh, or 0Ch) | | | | | | | |
| 1 | Reserved | | | | | | | CAT |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0014h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | | | SOURCE TARGET DESCRIPTOR INDEX | | | | |
| 5 | | | | | | | | (LSB) |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | Reserved | | | | | | | |
| 9 | (MSB) | | | STREAM DEVICE TRANSFER LENGTH | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | Reserved | | | | | | | |
| 13 | Reserved | | | | | | | |
| 14 | (MSB) | | | BLOCK DEVICE NUMBER OF BLOCKS | | | | |
| 15 | | | | | | | | (LSB) |
| 16 | (MSB) | | | BLOCK DEVICE LOGICAL BLOCK ADDRESS | | | | |
| 23 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Two DESCRIPTOR TYPE CODE values use the segment descriptor format shown in table 24 and described in this clause.

For descriptor type code 00h (block→stream), the copy manager shall copy the data from the source block device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination stream device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using the logical blocks starting at the location identified by the BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of blocks specified in the BLOCK DEVICE NUMBER OF BLOCKS field. The data shall be written to the stream device starting at the current position of the media.

For descriptor type code 0Bh (block→stream+application client), the copy manager shall copy the data from the source block device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination stream device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using the logical blocks starting at the location identified by the BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of blocks specified in the BLOCK DEVICE NUMBER OF BLOCKS field. The data shall be written to the stream device starting at the current position of the media. The copy manager also shall hold a copy of the read data for delivery to the application client upon completion of the EXTENDED COPY command in response to a RECEIVE COPY RESULTS command with RECEIVE DATA service action as described in 7.77.2. If the copy manager supports the 0Bh

descriptor type code it also shall support the RECEIVE COPY RESULTS command with RECEIVE DATA service action.

The CAT bit is described in 7.1.5.

The DESCRIPTOR LENGTH field shall contain 20 (0014h). The SOURCE TARGET DESCRIPTOR INDEX and DESTINATION TARGET DESCRIPTOR INDEX fields are described in 7.1.5.

The STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be written on each write operation to the stream device. See 7.1.4.7 for a description of how data in the STREAM DEVICE TRANSFER LENGTH field in the segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the sequential-access device type.

The BLOCK DEVICE NUMBER OF BLOCKS field specifies the number logical blocks to be read in the segment. A value of zero indicates that no blocks shall be transferred in this segment. This shall not be considered as an error.

The BLOCK DEVICE LOGICAL BLOCK ADDRESS field specifies the starting logical block address on the block device for this segment.

#### 7.1.5.2 Stream device to block device operations

The segment descriptor format shown in table 24 (see 7.1.5.1) also is used by the copy operations that move data from a stream device to a block device. Two DESCRIPTOR TYPE CODE values use the segment descriptor format shown in table 24 and described in this clause.

For descriptor type code 01h (stream→block), the copy manager shall copy the data from the source stream device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination block device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using the stream data starting at the current position of the stream device. The data shall be written to logical blocks starting at the location identified by the BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of blocks specified in the BLOCK DEVICE NUMBER OF BLOCKS field.

For descriptor type code 0Ch (stream→block+application client), the copy manager shall copy the data from the source stream device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination block device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using the stream data starting at the current position of the stream device. The data shall be written to logical blocks starting at the location identified by the BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of blocks specified in the BLOCK DEVICE NUMBER OF BLOCKS field. The copy manager also shall hold a copy of the read data for delivery to the application client upon completion of the EXTENDED COPY command in response to a RECEIVE COPY RESULTS command with RECEIVE DATA service action as described in 7.77.2. If the copy manager supports the 0Ch descriptor type code it also shall support the RECEIVE COPY RESULTS command with RECEIVE DATA service action.

The CAT bit is described in 7.1.5.

The DESCRIPTOR LENGTH field shall contain 20 (0014h). The SOURCE TARGET DESCRIPTOR INDEX and DESTINATION TARGET DESCRIPTOR INDEX fields are described in 7.1.5.

The STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be read from the source stream device on each read operation. See 7.1.4.7 for a description of how data in the STREAM DEVICE TRANSFER LENGTH field in the segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the sequential-access device type.

The BLOCK DEVICE NUMBER OF BLOCKS field specifies the number blocks to be written by the segment. A value of zero indicates that no blocks shall be transferred in this segment. This shall not be considered as an error.

The BLOCK DEVICE LOGICAL BLOCK ADDRESS field specifies the starting logical block address on the block device for this segment.

### 7.1.5.3 Block device to block device operations

The segment descriptor format shown in table 25 is used by the copy operations that move data from a block device to a block device.

**Table 25 — Block device to block device segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (02h or 0Dh) | | | | | | | |
| 1 | Reserved | | | | | | DC | CAT |
| 2 | (MSB) | | | | | | | |
| 3 | DESCRIPTOR LENGTH (0018h) | | | | | | | (LSB) |
| 4 | (MSB) | | | | | | | |
| 5 | SOURCE TARGET DESCRIPTOR INDEX | | | | | | | (LSB) |
| 6 | (MSB) | | | | | | | |
| 7 | DESTINATION TARGET DESCRIPTOR INDEX | | | | | | | (LSB) |
| 8 | Reserved | | | | | | | |
| 9 | Reserved | | | | | | | |
| 10 | (MSB) | | | | | | | |
| 11 | BLOCK DEVICE NUMBER OF BLOCKS | | | | | | | (LSB) |
| 12 | (MSB) | | | | | | | |
| 19 | SOURCE BLOCK DEVICE LOGICAL BLOCK ADDRESS | | | | | | | (LSB) |
| 20 | (MSB) | | | | | | | |
| 27 | DESTINATION BLOCK DEVICE LOGICAL BLOCK ADDRESS | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Two DESCRIPTOR TYPE CODE values use the segment descriptor format shown in table 25 and described in this clause.

For descriptor type code 02h (block→block), the copy manager shall copy the data from the source block device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination block device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using the logical blocks starting at the location identified by the source BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of blocks specified in the BLOCK DEVICE NUMBER OF BLOCKS field. The data shall be written to logical blocks starting at the location identified by the DESTINATION BLOCK DEVICE LOGICAL BLOCK ADDRESS field.

For descriptor type code 0Dh (block→block+application client), the copy manager shall copy the data from the source block device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination block device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using the logical blocks starting at the location identified by the source BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of blocks specified in the BLOCK DEVICE NUMBER OF BLOCKS field. The data shall be written to logical blocks starting at the location identified by the DESTINATION BLOCK DEVICE LOGICAL BLOCK ADDRESS field. The copy manager also shall hold a copy of the read data for delivery to the application client upon completion of the EXTENDED COPY command in response to a RECEIVE COPY RESULTS command with RECEIVE DATA service action as described

in 7.77.2. If the copy manager supports the 0Dh descriptor type code it also shall support the RECEIVE COPY RESULTS command with RECEIVE DATA service action.

The CAT bit is described in 7.1.5.

The destination count (DC) bit indicates whether the BLOCK DEVICE NUMBER OF BLOCKS field refers to the source or destination device. A DC bit of zero indicates that the BLOCK DEVICE NUMBER OF BLOCKS field refers to the source device. A DC bit of one indicates that the BLOCK DEVICE NUMBER OF BLOCKS field refers to the destination device.

The DESCRIPTOR LENGTH field shall contain 24 (0018h). The SOURCE TARGET DESCRIPTOR INDEX and DESTINATION TARGET DESCRIPTOR INDEX fields are described in 7.1.5.

The BLOCK DEVICE NUMBER OF BLOCKS field specifies the number of blocks to be transferred from the source block device to the destination block device. A value of zero indicates that no blocks are to be transferred. This shall not be considered as an error.

The SOURCE BLOCK DEVICE LOGICAL BLOCK ADDRESS field specifies the logical block address from which the reading of data will start.

The DESTINATION BLOCK DEVICE LOGICAL BLOCK ADDRESS field specifies the logical block address to which the writing of data will begin.

### 7.1.5.4 Stream device to stream device operations

The segment descriptor format shown in table 26 is used by the copy operations that move data from a stream device to a stream device.

**Table 26 — Stream device to stream device segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (03h or 0Eh) | | | | | | | |
| 1 | Reserved | | | | | | DC | CAT |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0010h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | | | SOURCE TARGET DESCRIPTOR INDEX | | | | |
| 5 | | | | | | | | (LSB) |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | Reserved | | | | | | | |
| 9 | (MSB) | | | SOURCE STREAM DEVICE TRANSFER LENGTH | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | Reserved | | | | | | | |
| 13 | (MSB) | | | DESTINATION STREAM DEVICE TRANSFER LENGTH | | | | |
| 14 | | | | | | | | |
| 15 | | | | | | | | (LSB) |
| 16 | (MSB) | | | TRANSFER COUNT | | | | |
| 19 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Two DESCRIPTOR TYPE CODE values use the segment descriptor format shown in table 26 and described in this clause.

For descriptor type code 03h (stream→stream), the copy manager shall copy the data from the source stream device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination stream device identified by the DESTINATION TARGET DESCRIPTOR INDEX field. Data shall be read from the source stream device starting at the current position of the source stream device. Data shall be written to the destination stream device starting at the current position of the destination stream device. The TRANSFER COUNT and DC fields define the number of read or write operations to be performed by the copy manager.

For descriptor type code 0Eh (stream→stream+application client), the copy manager shall copy the data from the source stream device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination stream device identified by the DESTINATION TARGET DESCRIPTOR INDEX field. Data shall be read from the source stream device starting at the current position of the source stream device. Data shall be written to the destination stream device starting at the current position of the destination stream device. The TRANSFER COUNT and DC fields define the number of read or write operations to be performed by the copy manager. The copy manager also shall hold a copy of the read data for delivery to the application client upon completion of the EXTENDED COPY command in response to a RECEIVE COPY RESULTS command with RECEIVE DATA service action as described in 7.77.2. If

the copy manager supports the 0Eh descriptor type code it also shall support the RECEIVE COPY RESULTS command with RECEIVE DATA service action.

The CAT bit is described in 7.1.5.

The destination count (DC) bit indicates whether the TRANSFER COUNT field refers to the source or destination device. A DC bit of zero indicates that the TRANSFER COUNT field indicates the number of read operations to be performed on the source device. A DC bit of one indicates that the TRANSFER COUNT field indicates the number of write operations to be performed on the destination device.

The DESCRIPTOR LENGTH field shall contain 16 (0010h). The SOURCE TARGET DESCRIPTOR INDEX and DESTINATION TARGET DESCRIPTOR INDEX fields are described in 7.1.5.

The source STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be read from the source stream device on each read operation. See 7.1.4.7 for a description of how data in the SOURCE STREAM DEVICE TRANSFER LENGTH field in the segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the source sequential-access device type.

The DESTINATION STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be written to the destination stream device on each write operation. See 7.1.4.7 for a description of how data in the DESTINATION STREAM DEVICE TRANSFER LENGTH field in the segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the destination sequential-access device type.

The TRANSFER COUNT field specifies the number of read/write operations that shall be executed for this segment descriptor. A value of zero indicates that no reads or writes are to be performed. This shall not be considered as an error. The DC bit indicates whether the number of reads or writes is specified.

### 7.1.5.5 Inline data to stream device operation

The segment descriptor format shown in table 27 instructs the copy manager to write inline data from the EXTENDED COPY parameter list to a stream device.

**Table 27 — Inline data to stream device segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (04h) | | | | | | | |
| 1 | Reserved | | | | | | | CAT |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0010h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | Reserved | | | | | | | |
| 5 | Reserved | | | | | | | |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | Reserved | | | | | | | |
| 9 | (MSB) | | | STREAM DEVICE TRANSFER LENGTH | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | (MSB) | | | INLINE DATA OFFSET | | | | |
| 15 | | | | | | | | (LSB) |
| 16 | (MSB) | | | INLINE DATA NUMBER OF BYTES | | | | |
| 19 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 04h (inline→stream) instructs the copy manager to write inline data from the EXTENDED COPY parameter list to a stream device. The inline data shall be read from the optional inline data at the end of the EXTENDED COPY parameter list. The data shall be written to the destination stream device identified by the DESTINATION TARGET DESCRIPTOR INDEX field starting at the current position of the stream device.

The CAT bit is described in 7.1.5.

The DESCRIPTOR LENGTH field shall contain 16 (0010h). The DESTINATION TARGET DESCRIPTOR INDEX field is described in 7.1.5.

The STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be written to the stream device on each write operation. See 7.1.4.7 for a description of how data in the STREAM DEVICE TRANSFER LENGTH field in the segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the destination sequential-access device type.

The value in the INLINE DATA OFFSET field is added to the location of the first byte of inline data in the EXTENDED COPY parameter list (see table 9) to locate the first byte of inline data to be written to the stream device. The INLINE DATA OFFSET value shall be a multiple of 4.

The INLINE DATA NUMBER OF BYTES field specifies the number of bytes of inline data that are to be transferred to the stream device.

If the sum of the INLINE DATA OFFSET and the INLINE DATA NUMBER OF BYTES values exceeds the value in the INLINE DATA LENGTH field (see table 9), the copy manager shall terminate the command with a CHECK CONDITION status. The sense key shall be set to COPY ABORTED and the additional sense code shall be set to INLINE DATA LENGTH EXCEEDED (26h/0Bh).

### 7.1.5.6 Embedded data to stream device operation

The segment descriptor format shown in table 28 instructs the copy manager to write embedded data from the segment descriptor to a stream device.

**Table 28 — Embedded data to stream device segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (05h) | | | | | | | |
| 1 | Reserved | | | | | | | CAT |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (n-4) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | Reserved | | | | | | | |
| 5 | Reserved | | | | | | | |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | Reserved | | | | | | | |
| 9 | (MSB) | | | STREAM DEVICE TRANSFER LENGTH | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | (MSB) | | | EMBEDDED DATA NUMBER OF BYTES | | | | |
| 13 | | | | | | | | (LSB) |
| 14 | | | | Reserved | | | | |
| 15 | | | | | | | | |
| 16 | EMBEDDED DATA | | | | | | | |
| n | | | | | | | | |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 05h (embedded→stream) instructs the copy manager to write embedded data from the segment descriptor to a stream device. The embedded data shall be read from the segment descriptor. The data shall be written to the destination stream device identified by the DESTINATION TARGET DESCRIPTOR INDEX field starting at the current position of the stream device.

The CAT bit is described in 7.1.5.

The DESCRIPTOR LENGTH field shall contain the length in bytes of the fields that follow the DESCRIPTOR LENGTH field (including embedded data). The value in the DESCRIPTOR LENGTH field shall be a multiple of 4.

The DESTINATION TARGET DESCRIPTOR INDEX field is described in 7.1.5.

The STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be written to the stream device on each write operation. See 7.1.4.7 for a description of how data in the STREAM DEVICE TRANSFER LENGTH field in the segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the destination sequential-access device type.

The EMBEDDED DATA NUMBER OF BYTES field specifies the number of bytes of embedded data that are to be transferred to the stream device. The EMBEDDED DATA NUMBER OF BYTES value shall be less than or equal to the DESCRIPTOR LENGTH value minus 12.

### 7.1.5.7 Stream device to discard operation

The segment descriptor format shown in table 29 instructs the copy manager to read data from a stream device and not copy it to any destination device.

**Table 29 — Stream device to discard segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (06h or 0Fh) | | | | | | | |
| 1 | Reserved | | | | | | | CAT |
| 2 | (MSB) | DESCRIPTOR LENGTH (000Ch) | | | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | SOURCE TARGET DESCRIPTOR INDEX | | | | | | |
| 5 | | | | | | | | (LSB) |
| 6 | Reserved | | | | | | | |
| 7 | Reserved | | | | | | | |
| 8 | Reserved | | | | | | | |
| 9 | (MSB) | STREAM DEVICE TRANSFER LENGTH | | | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | (MSB) | NUMBER OF BYTES | | | | | | |
| 15 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Two DESCRIPTOR TYPE CODE values use the segment descriptor format shown in table 29 and described in this clause.

For descriptor type code 06h (stream→discard), the copy manager shall read data from the source stream device identified by the SOURCE TARGET DESCRIPTOR INDEX field starting at the current position of the source stream device. The data read shall not be transferred to any destination device.

For descriptor type code 0Fh (stream→discard+application client), the copy manager shall read data from the source stream device identified by the SOURCE TARGET DESCRIPTOR INDEX field starting at the current position of the source stream device and hold a copy of the read data for delivery to the application client upon completion of the EXTENDED COPY command in response to a RECEIVE COPY RESULTS command with RECEIVE DATA service action as described in 7.77.2. If the copy manager supports the 0Fh descriptor type code it also shall support the RECEIVE COPY RESULTS command with RECEIVE DATA service action.

The CAT bit is described in 7.1.5.

The DESCRIPTOR LENGTH field shall contain 12 (000Ch). The DESTINATION TARGET DESCRIPTOR INDEX field is described in 7.1.5.

The source STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be read from the source stream device on each read operation. See 7.1.4.7 for a description of how data in the SOURCE STREAM DEVICE TRANSFER LENGTH field in the segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the source sequential-access device type.

The NUMBER OF BYTES field specifies the number of bytes to be read from the stream device.

### 7.1.5.8 Verify device operation

The segment descriptor format shown in table 30 instructs the copy manager to verify the accessibility of a device.

**Table 30 — Verify device operation segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (07h) | | | | | | | |
| 1 | Reserved | | | | | | | |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0004h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | | | SOURCE TARGET DESCRIPTOR INDEX | | | | |
| 5 | | | | | | | | (LSB) |
| 6 | Reserved | | | | | | | |
| 7 | Reserved | | | | | | | |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 07h instructs the copy manager to verify the accessibility of the device identified by the SOURCE TARGET DESCRIPTOR INDEX field. The accessibility should be verified without disturbing established unit attention or ACA conditions, for example, using the INQUIRY command (see 7.4).

The DESCRIPTOR LENGTH field shall contain 4 (0004h). The SOURCE TARGET DESCRIPTOR INDEX field is described in 7.1.5.

### 7.1.5.9 Block device with offset to stream device operation

The segment descriptor format shown in table 31 is used to instruct the copy manager to move data from a block device with a byte offset to a stream device or vice versa.

**Table 31 — Block device with offset to or from stream device segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (08h or 09h) | | | | | | | |
| 1 | Reserved | | | | | | | CAT |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0018h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | | | SOURCE TARGET DESCRIPTOR INDEX | | | | |
| 5 | | | | | | | | (LSB) |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | Reserved | | | | | | | |
| 9 | (MSB) | | | STREAM DEVICE TRANSFER LENGTH | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | (MSB) | | | NUMBER OF BYTES | | | | |
| 15 | | | | | | | | (LSB) |
| 16 | (MSB) | | | BLOCK DEVICE LOGICAL BLOCK ADDRESS | | | | |
| 23 | | | | | | | | (LSB) |
| 24 | Reserved | | | | | | | |
| 25 | Reserved | | | | | | | |
| 26 | (MSB) | | | BLOCK DEVICE BYTE OFFSET | | | | |
| 27 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 08h (block<o>→stream) instructs the copy manager to copy the data from the source block device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination stream device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using data starting at the location identified by BLOCK DEVICE BYTE OFFSET field in the logical block identified by the BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of bytes specified in the NUMBER OF BYTES field. The data shall be written to the stream device starting at the current position of the media.

The CAT bit is described in 7.1.5.

The DESCRIPTOR LENGTH field shall contain 24 (0018h). The SOURCE TARGET DESCRIPTOR INDEX and DESTINATION TARGET DESCRIPTOR INDEX fields are described in 7.1.5.

The STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be written on each write operation to the stream device. See 7.1.4.7 for a description of how data in the STREAM DEVICE TRANSFER LENGTH field in the

segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the sequential-access device type.

The NUMBER OF BYTES field specifies the number bytes to be read. A value of zero indicates that no bytes shall be transferred in this segment. This shall not be considered as an error.

The BLOCK DEVICE LOGICAL BLOCK ADDRESS field specifies the starting logical block address on the source block device for this segment.

The BLOCK DEVICE BYTE OFFSET field specifies the offset into the first source block at which to begin reading bytes.

**7.1.5.10 Stream device to block device with offset operation**

The segment descriptor format shown in table 31 (see 7.1.5.9) also is used to instruct the copy manager to move data from a stream device to a block device with a byte offset.

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 09h (stream→block<o>) instructs the copy manager to copy the data from the source stream device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination block device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using the stream data starting at the current position of the stream device. The data shall be written starting at the location identified by BLOCK DEVICE BYTE OFFSET field in the logical block identified by the BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of bytes specified in the NUMBER OF BYTES field.

The content of the starting logical block on the destination device before the starting offset shall be preserved. The content on the ending logical block beyond the end of the transfer shall be preserved. The copy manager may implement this operation by reading the starting and ending logical blocks, modifying a portion of the blocks as required, and writing the full blocks to the destination device.

The CAT bit is described in 7.1.5.

The DESCRIPTOR LENGTH field shall contain 24 (0018h). The SOURCE TARGET DESCRIPTOR INDEX and DESTINATION TARGET DESCRIPTOR INDEX fields are described in 7.1.5.

The STREAM DEVICE TRANSFER LENGTH field specifies the amount of data to be written on each write operation to the stream device. See 7.1.4.7 for a description of how data in the STREAM DEVICE TRANSFER LENGTH field in the segment descriptor interacts with data in the STREAM BLOCK LENGTH field in the device type specific target descriptor parameters for the sequential-access device type.

The NUMBER OF BYTES field specifies the number bytes to be read. A value of zero indicates that no bytes shall be transferred in this segment. This shall not be considered as an error.

The BLOCK DEVICE LOGICAL BLOCK ADDRESS field specifies the starting logical block address on the destination block device for this segment.

The BLOCK DEVICE BYTE OFFSET field is the offset into the first destination block at which to begin writing data to the destination block device.

### 7.1.5.11 Block device with offset to block device with offset operation

The segment descriptor format shown in table 32 instructs the copy manager to move data from a block device with a byte offset to a block device with a byte offset.

**Table 32 — Block device with offset to block device with offset segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (0Ah) | | | | | | | |
| 1 | Reserved | | | | | | | CAT |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (001Ch) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | | | SOURCE TARGET DESCRIPTOR INDEX | | | | |
| 5 | | | | | | | | (LSB) |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | (MSB) | | | NUMBER OF BYTES | | | | |
| 11 | | | | | | | | (LSB) |
| 12 | (MSB) | | | SOURCE BLOCK DEVICE LOGICAL BLOCK ADDRESS | | | | |
| 19 | | | | | | | | (LSB) |
| 20 | (MSB) | | | DESTINATION BLOCK DEVICE LOGICAL BLOCK ADDRESS | | | | |
| 27 | | | | | | | | (LSB) |
| 28 | (MSB) | | | SOURCE BLOCK DEVICE BYTE OFFSET | | | | |
| 29 | | | | | | | | (LSB) |
| 30 | (MSB) | | | DESTINATION BLOCK DEVICE BYTE OFFSET | | | | |
| 31 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 0Ah (block<o>→block<o>) instructs the copy manager to copy the data from the source block device identified by the SOURCE TARGET DESCRIPTOR INDEX field to the destination block device identified by the DESTINATION TARGET DESCRIPTOR INDEX field using data starting at the location identified by SOURCE BLOCK DEVICE BYTE OFFSET field in the logical block identified by the SOURCE BLOCK DEVICE LOGICAL BLOCK ADDRESS field and continuing for the number of bytes specified in the NUMBER OF BYTES field. The data shall be written starting at the location identified by DESTINATION BLOCK DEVICE BYTE OFFSET field in the logical block identified by the DESTINATION BLOCK DEVICE LOGICAL BLOCK ADDRESS field.

The content of the starting logical block on the destination device before the starting offset shall be preserved. The content on the ending logical block beyond the end of the transfer shall be preserved. The copy manager may implement this operation by reading the starting and ending logical blocks, modifying a portion of the blocks as required, and writing the full blocks to the destination device.

The CAT bit is described in 7.1.5.

The DESCRIPTOR LENGTH field shall contain 28 (001Ch). The SOURCE TARGET DESCRIPTOR INDEX and DESTINATION TARGET DESCRIPTOR INDEX fields are described in 7.1.5.

The NUMBER OF BYTES field specifies the number bytes to be read. A value of zero indicates that no bytes shall be transferred in this segment. This shall not be considered as an error.

The SOURCE BLOCK DEVICE LOGICAL BLOCK ADDRESS field specifies the starting address on the source block device for this segment.

The DESTINATION BLOCK DEVICE LOGICAL BLOCK ADDRESS field specifies the starting logical block address on the destination block device for this segment.

The SOURCE BLOCK DEVICE BYTE OFFSET field specifies the offset into the first source block at which to begin reading bytes.

The DESTINATION BLOCK DEVICE BYTE OFFSET field is the offset into the first destination block at which to begin writing data to the destination block device.

### 7.1.5.12 Write filemarks operation

The segment descriptor format shown in table 33 instructs the copy manager to write filemarks or setmarks on the destination tape device.

**Table 33 — Write filemarks operation segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (10h) | | | | | | | |
| 1 | Reserved | | | | | | | |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0008h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | Reserved | | | | | | | |
| 5 | Reserved | | | | | | | |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | Reserved | | | | | | WSMK | Reserved |
| 9 | (MSB) | | | TRANSFER LENGTH | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 10h (filemark→tape) instructs the copy manager to write filemarks or setmarks to the destination tape device identified by the DESTINATION TARGET DESCRIPTOR INDEX field starting at the current position of the tape device. If the PERIPHERAL DEVICE TYPE field in the target descriptor identified by the DESTINATION TARGET DESCRIPTOR INDEX field does not contain 01h, the copy manager shall terminate the command with a CHECK CONDITION status. The sense key shall be set to COPY ABORTED and the additional sense code shall be set to INVALID OPERATION FOR COPY SOURCE OR DESTINATION (26h/0Ch).

The DESCRIPTOR LENGTH field shall contain 8 (0008h). The DESTINATION TARGET DESCRIPTOR INDEX field is described in 7.1.5.

If the write setmark (WSMK) bit is one, the TRANSFER LENGTH field specifies the number of setmarks to be written. If the WSMK bit is zero, the TRANSFER LENGTH field specifies the number of filemarks to be written.

### 7.1.5.13 Space operation

The segment descriptor format shown in table 34 instructs the copy manager to send a SPACE command (see SSC) to the destination tape device.

**Table 34 — Space operation segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (11h) | | | | | | | |
| 1 | Reserved | | | | | | | |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0008h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | Reserved | | | | | | | |
| 5 | Reserved | | | | | | | |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | Reserved | | | | | CODE | | |
| 9 | (MSB) | | | COUNT | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 11h (space→tape) instructs the copy manager to send a SPACE command to the destination tape device identified by the DESTINATION TARGET DESCRIPTOR INDEX field. If the PERIPHERAL DEVICE TYPE field in the target descriptor identified by the DESTINATION TARGET DESCRIPTOR INDEX field does not contain 01h, the copy manager shall terminate the command with a CHECK CONDITION status. The sense key shall be set to COPY ABORTED and the additional sense code shall be set to INVALID OPERATION FOR COPY SOURCE OR DESTINATION (26h/0Ch).

The DESCRIPTOR LENGTH field shall contain 8 (0008h). The DESTINATION TARGET DESCRIPTOR INDEX field is described in 7.1.5.

The CODE and COUNT field contents in the SPACE command sent to the destination tape device shall be copied from the CODE and COUNT fields in the segment descriptor. All other fields in the SPACE command sent to the destination tape device that affect the positioning of the tape shall be set to zero.

### 7.1.5.14 Locate operation

The segment descriptor format shown in table 35 instructs the copy manager to send a LOCATE command (see SSC) to the destination tape device.

**Table 35 — Locate operation segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (12h) | | | | | | | |
| 1 | Reserved | | | | | | | |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0008h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | Reserved | | | | | | | |
| 5 | Reserved | | | | | | | |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | (MSB) | | | BLOCK ADDRESS | | | | |
| 11 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 12h (locate→tape) instructs the copy manager to send a LOCATE command to the destination tape device identified by the DESTINATION TARGET DESCRIPTOR INDEX field. If the PERIPHERAL DEVICE TYPE field in the target descriptor identified by the DESTINATION TARGET DESCRIPTOR INDEX field does not contain 01h, the copy manager shall terminate the command with a CHECK CONDITION status. The sense key shall be set to COPY ABORTED and the additional sense code shall be set to INVALID OPERATION FOR COPY SOURCE OR DESTINATION (26h/0Ch).

The DESCRIPTOR LENGTH field shall contain 8 (0008h). The DESTINATION TARGET DESCRIPTOR INDEX field is described in 7.1.5.

The BLOCK ADDRESS field contents in the LOCATE command sent to the destination tape device shall be copied from the BLOCK ADDRESS field in the segment descriptor. All other fields in the LOCATE command sent to the destination tape device that affect the positioning of the tape shall be set to zero.

NOTE 5 The restrictions described above for the LOCATE command limit the operation to locating SCSI logical block addresses in the current tape partition.

### 7.1.5.15 Tape device image copy operation

The segment descriptor format shown in table 36 instructs the copy manager to perform an image copy from the source tape device to the destination tape device.

**Table 36 — Tape device image copy segment descriptor**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | DESCRIPTOR TYPE CODE (13h) | | | | | | | |
| 1 | Reserved | | | | | | | |
| 2 | (MSB) | | | DESCRIPTOR LENGTH (0008h) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | (MSB) | | | SOURCE TARGET DESCRIPTOR INDEX | | | | |
| 5 | | | | | | | | (LSB) |
| 6 | (MSB) | | | DESTINATION TARGET DESCRIPTOR INDEX | | | | |
| 7 | | | | | | | | (LSB) |
| 8 | (MSB) | | | COUNT | | | | |
| 11 | | | | | | | | (LSB) |

The DESCRIPTOR TYPE CODE field is described in 7.1.3 and 7.1.5. Descriptor type code 13h (<i>tape→<i>tape) instructs the copy manager to create a compatible image of the source device medium identified by the SOURCE TARGET DESCRIPTOR INDEX field on the destination device medium identified by the DESTINATION TARGET DESCRIPTOR INDEX field beginning at their current positions. If the PERIPHERAL DEVICE TYPE field in the target descriptor identified by the SOURCE TARGET DESCRIPTOR INDEX field or the DESTINATION TARGET DESCRIPTOR INDEX field does not contain 01h, the copy manager shall terminate the command with a CHECK CONDITION status. The sense key shall be set to COPY ABORTED and the additional sense code shall be set to INVALID OPERATION FOR COPY SOURCE OR DESTINATION (26h/0Ch).

The DESCRIPTOR LENGTH field shall contain 8 (0008h). The SOURCE TARGET DESCRIPTOR INDEX and DESTINATION TARGET DESCRIPTOR INDEX fields are described in 7.1.5.

The tape image copy operation terminates when:

a) the source device encounters an end-of-partition as defined by the source device;
b) the source device encounters an end-of-data as defined by the source device (i.e., BLANK CHECK sense key);
c) the copy manager has copied the number of consecutive filemarks specified in the COUNT field from the source device to the destination device; or
d) the copy manager has copied the number of consecutive filemarks and/or setmarks specified in the COUNT field from the source device to the destination device, if the RSMK bit in the device configuration page (see SSC) of the source device is one.

The COUNT field of zero indicates that the EXTENDED COPY command shall not terminate due to any number of consecutive filemarks or setmarks. Other error or exception conditions (e.g., early-warning, end-of-partition on destination device) may cause the EXTENDED COPY command to terminate prior to completion. In such cases, it is not possible to calculate a residue, so the INFORMATION field in the sense data shall be set to zero.

## 7.77 RECEIVE COPY RESULTS command

The RECEIVE COPY RESULTS command (see table 37) provides a means for the application client to receive information about the copy manager or the results of a previous (or current) EXTENDED COPY command (see 7.1).

**Table 37 — RECEIVE COPY RESULTS command**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | OPERATION CODE (84h) | | | | | | | |
| 1 | Reserved | | | SERVICE ACTION | | | | |
| 2 | LIST IDENTIFIER | | | | | | | |
| 3 | Reserved | | | | | | | |
| 4 | Reserved | | | | | | | |
| 5 | Reserved | | | | | | | |
| 6 | Reserved | | | | | | | |
| 7 | Reserved | | | | | | | |
| 8 | Reserved | | | | | | | |
| 9 | Reserved | | | | | | | |
| 10 | (MSB) | | | ALLOCATION LENGTH | | | | |
| 11 | | | | | | | | |
| 12 | | | | | | | | |
| 13 | | | | | | | | (LSB) |
| 14 | Reserved | | | | | | | |
| 15 | CONTROL | | | | | | | |

The service actions defined for the RECEIVE COPY RESULTS command are shown in table 38.

**Table 38 — RECEIVE COPY RESULTS service action codes**

| Code | Name | Description | Returns Data |
|---|---|---|---|
| 00h | COPY STATUS | Return the current copy status of the EXTENDED COPY command identified by the LIST IDENTIFIER field. | Immediately |
| 01h | RECEIVE DATA | Return the held data read by EXTENDED COPY command identified by the LIST IDENTIFIER field. | When identified command has completed |
| 03h | OPERATING PARAMETERS | Return copy manager operating parameters. | Immediately |
| 04h-1Fh | Reserved. | | |

The LIST IDENTIFIER field identifies the EXTENDED COPY command (see 7.1) about which information is desired. The RECEIVE COPY RESULTS command shall return information from the EXTENDED COPY command received from the same initiator with a list identifier that matches the list identifier specified in the RECEIVE COPY RESULTS CDB. If no EXTENDED COPY command known to the copy manager has a matching list identifier, then

the command shall be terminated with CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST, and the additional sense code shall be set to INVALID FIELD IN CDB.

If the LIST IDENTIFIER field identifies the EXTENDED COPY command that had the NRCR bit set to 1 in the parameter data (see 7.1), the copy manager may respond to a RECEIVE COPY RESULTS command in the same manner it would if the EXTENDED COPY command had never been received.

The actual length of the RECEIVE COPY RESULTS parameter data is available in the AVAILABLE DATA parameter data field. The ALLOCATION LENGTH field in the CDB indicates how much space has been reserved for the returned parameter list. If the length is not sufficient to contain the entire parameter list, the first portion of the list shall be returned. This shall not be considered an error. If the remainder of the list is required, the application client should send a new RECEIVE COPY RESULTS command with a ALLOCATION LENGTH field large enough to contain the entire parameter list.

### 7.77.1 COPY STATUS service action

In response to the COPY STATUS service action, the copy manager shall return the current status of the EXTENDED COPY command (see 7.1) identified by the LIST IDENTIFIER field in the CDB. Table 40 shows the format of the information returned by the copy manager in response to the COPY STATUS service action. If a device server supports the EXTENDED COPY command, then it shall also support the RECEIVE COPY RESULTS command with COPY STATUS service action.

**Table 39 — Parameter data for the COPY STATUS service action**

| Bit<br>Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | (MSB) | | | | | | | |
| 3 | | | | AVAILABLE DATA (00000008h) | | | | (LSB) |
| 4 | | | | STATUS | | | | |
| 5 | (MSB) | | | | | | | |
| 6 | | | | SEGMENTS PROCESSED | | | | (LSB) |
| 7 | | | | TRANSFER COUNT UNITS | | | | |
| 8 | (MSB) | | | | | | | |
| 11 | | | | TRANSFER COUNT | | | | (LSB) |

After completion of an EXTENDED COPY command, the copy manager shall preserve all data returned by a COPY STATUS service action for a vendor specific period of time. The copy manager shall discard the COPY STATUS data when:

a) a RECIEVE COPY RESULTS command with COPY STATUS service action is received from the same initiator with a matching list identifier;
b) when another EXTENDED COPY command is received from the same initiator using the same list identifier;
c) when the copy manager detects a hard reset condition; or
d) when the copy manager requires the resources used to preserve the data.

The AVAILABLE DATA field shall contain the number of bytes present in the parameter data that follows, eight.

The STATUS field contains the current status of the EXTENDED COPY command identified by the LIST IDENTIFIER field in the CDB as defined in table 40.

**Table 40 — COPY STATUS STATUS values**

| STATUS value | Meaning |
|---|---|
| 00h | Operating in progress |
| 01h | Operation completed without errors |
| 02h | Operation completed with errors |
| 04h - FFh | Reserved |

The SEGMENTS PROCESSED field contains the number of segments the copy manager has processed for the EXTENDED COPY command identified by the LIST IDENTIFIER field in the CDB including the segment currently being processed. This field shall be zero if the copy manager has not yet begun processing segment descriptors.

The TRANSFER COUNT UNITS field specifies the units for the TRANSFER COUNT field as defined in table 41.

**Table 41 — COPY STATUS TRANSFER COUNT UNITS values**

| Value | Meaning | Multiplier to convert TRANSFER COUNT field to bytes |
|---|---|---|
| 00h | Bytes | 1 |
| 01h | Kilo-bytes | $2^{10}$ or 1024 |
| 02h | Mega-bytes | $2^{20}$ |
| 03h | Giag-bytes | $2^{30}$ |
| 04h | Tera-bytes | $2^{40}$ |
| 05h | Peta-bytes | $2^{50}$ |
| 06h - FFh | Reserved | |

The TRANSFER COUNT field specifies the amount of data written to a destination device for the EXTENDED COPY command identified by the LIST IDENTIFIER field in the CDB prior to receiving the RECEIVE COPY RESULTS command with COPY STATUS service action.

### 7.77.2 RECEIVE DATA service action

If the copy manager supports those segment descriptors that hold read data for transfer to the application client, then the RECEIVE DATA service action causes the copy manager to return the held data using the format shown in table 42. If a copy manager supports any of the segment descriptor type codes that require read data to be held for the application client (see 7.1.3), then it shall also support the RECEIVE COPY RESULTS command with RECEIVE DATA service action.

**Table 42 — Parameter data for the RECEIVE DATA service action**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | (MSB) | | | AVAILABLE DATA (n-4) | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | | | | Held data | | | | |
| n | | | | | | | | |

Following completion of an EXTENDED COPY command, the copy manager shall preserve all data returned by a RECIEVE DATA service action for a vendor specific period of time. The application client should issue a RECEIVE COPY RESULTS command with RECEIVE DATA service action immediately following completion of the EXTENDED COPY command to insure that the data is not discarded by the copy manager. The copy manager shall discard the buffered inline data:

a) after all data held for a specific EXTENDED COPY command has been successfully transferred to the application client;
b) when a RECIEVE COPY RESULTS command with RECEIVE DATA service action has been received from the same initiator with a matching list identifier, with the ALLOCATION LENGTH field set to 0;
c) when another EXTENDED COPY command is received from the same initiator using the same list identifier;
d) when the copy manager detects a hard reset condition; or
e) when the copy manager requires the resources used to preserve the data.

The AVAILABLE DATA field shall contain the number of bytes of held data available for delivery to the application client. If the amount of held data sent to the application client is reduced due to insufficient allocation length, the AVAILABLE DATA field shall not be altered and the held data shall not be discarded.

The held data is the data held by the copy manager for delivery to the application client as proscribed by several segment descriptor type codes. The oldest byte read and held is returned in byte 4 and the byte most recently read and held is returned in byte n.

The maximum number of bytes that the copy manager will hold for the application client is returned in the HELD DATA LIMIT field in the copy manager's operating parameters (see 7.77.3). If the processing of segment descriptors requires more data to be held, the copy manager shall discard the oldest held data bytes to accommodate the new read data. When making room for new read data, the copy manager may discard more old data bytes than are needed immediately, but at any one time the copy manager shall never discard more than the smaller of 64 times the HELD DATA GRANULARITY value (see 7.77.3) or one quarter of the HELD DATA LIMIT value. Discarding of held data bytes shall not be considered an error. The only way the application client may detect the possible discarding of held data bytes is to compare the AVAILABLE DATA field to the HELD DATA LIMIT field.

### 7.77.3 OPERATING PARAMETERS service action

In response to the OPERATING PARAMETERS service action, the copy manager shall return its operating parameter information in the format shown in table 43. If a device server supports the EXTENDED COPY command (see 7.1), then it shall also support the RECEIVE COPY RESULTS command with OPERATING PARAMETERS service action.

**Table 43 — Parameter data for the OPERATING PARAMETERS service action**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | (MSB) | | | | | | | |
| 3 | | | | AVAILABLE DATA (n-4) | | | | (LSB) |
| 4 | (MSB) | | | | | | | |
| 5 | | | | MAXIMUM TARGET DESCRIPTOR COUNT | | | | (LSB) |
| 6 | (MSB) | | | | | | | |
| 7 | | | | MAXIMUM SEGMENT DESCRIPTOR COUNT | | | | (LSB) |
| 8 | (MSB) | | | | | | | |
| 11 | | | | MAXIMUM DESCRIPTOR LIST LENGTH | | | | (LSB) |
| 12 | (MSB) | | | | | | | |
| 15 | | | | MAXIMUM SEGMENT LENGTH | | | | (LSB) |
| 16 | (MSB) | | | | | | | |
| 19 | | | | MAXIMUM INLINE DATA LENGTH | | | | (LSB) |
| 20 | (MSB) | | | | | | | |
| 23 | | | | HELD DATA LIMIT | | | | (LSB) |
| 24 | MAXIMUM CONCURRENT COPIES | | | | | | | |
| 25 | DATA SEGMENT GRANULARITY (log 2) | | | | | | | |
| 26 | INLINE DATA GRANULARITY (log 2) | | | | | | | |
| 27 | HELD DATA GRANULARITY (log 2) | | | | | | | |
| 28 | Reserved | | | | | | | |
| 29 | Reserved | | | | | | | |
| 30 | Reserved | | | | | | | |
| 31 | IMPLEMENTED DESCRIPTOR LIST LENGTH (n-36) | | | | | | | |
| 32 | | | | | | | | |
| n | List of implemented descriptor type codes (ordered) | | | | | | | |

The AVAILABLE DATA field shall contain the length of the total length of the parameter data minus 4.

The MAXIMUM TARGET COUNT field contains the maximum number of target descriptors that the copy manager allows in a single EXTENDED COPY target descriptor list.

The MAXIMUM SEGMENT COUNT field contains the maximum number of segment descriptors that the copy manager allows in a single EXTENDED COPY segment descriptor list.

The MAXIMUM DESCRIPTOR LIST LENGTH field contains the maximum length, in bytes, of the target descriptor list and segment descriptor list. This length includes the embedded data but excludes inline data that follows the descriptors.

The MAXIMUM SEGMENT LENGTH field indicates the length, in bytes, of the largest amount of data that the copy manager supports writing via a single segment. Bytes introduced as a result of the pad bit being set to 1 (see 7.1.5) are not counted towards this limit. A value of zero indicates that the copy manager places no limits on the amount of data written by a single segment.

The MAXIMUM INLINE DATA LENGTH field indicates the length, in bytes, of the largest amount of inline data that copy manager supports in the EXTENDED COPY parameter list. This does not include data included as embedded data within the segment descriptors. The MAXIMUM INLINE DATA LENGTH field applies only to segment descriptors containing the 04h descriptor type code (see 7.1.5.5). The field shall be set to zero when the 04h descriptor type code is not supported by the copy manager.

The HELD DATA LIMIT field indicates the length, in bytes, of the largest amount of data the copy manager is capable of holding for return to the application client via the RECEIVE COPY RESULTS command with RECEIVE DATA service action (see 7.77.2).

The MAXIMUM CONCURRENT COPIES field contains the maximum number of EXTENDED COPY commands supported for concurrent processing by the copy manager.

The DATA SEGMENT GRANULARITY field indicates the length of the smallest data block that copy manager permits in a non-inline segment descriptor (segment descriptors with type codes other than 04h). The amount of data transferred by a single segment descriptor shall be a multiple of the granularity. The DATA SEGMENT GRANULARITY value is expressed as a power of 2. Bytes introduced as a result of the pad bit being set to 1 (see 7.1.5) are not counted towards the data length granularity.

The INLINE DATA GRANULARITY field indicates the length of the of the smallest block of inline data that the copy manager permits being written by a segment descriptor containing the 04h descriptor type code (see 7.1.5.5). The amount of inline data written by a single segment descriptor shall be a multiple of the granularity. The INLINE DATA GRANULARITY value is expressed as a power of 2. Bytes introduced as a result of the PAD bit being set to 1 (see 7.1.5) are not counted towards the length granularity.

If the copy manager encounters a data or inline segment descriptor that violates either the data segment granularity or the inline data granularity, the EXTENDED COPY command shall be terminated with a CHECK CONDITION status. The sense key shall be set to COPY ABORTED and the additional sense code shall be set to COPY SEGMENT GRANULARITY VIOLATION (26h/0Dh).

The HELD DATA GRANULARITY field indicates the length of the smallest block of held data that the copy manager shall transfer to the application client in response to a RECEIVE COPY RESULTS command with RECEIVE DATA service action (see 7.77.2). The amount of data held by the copy manager in response to any one segment descriptor shall be a multiple of this granularity. The HELD DATA GRANULARITY value is expressed as a power of 2.

The IMPLEMENTED DESCRIPTOR LIST LENGTH field contains the length, in bytes, of the List of implemented descriptor type codes.

The List of implemented descriptor type codes contains one byte for each segment or target DESCRIPTOR TYPE CODE value (see 7.1.3) supported by the copy manager, with a unique supported DESCRIPTOR TYPE CODE value in each byte. The DESCRIPTOR TYPE CODE values shall appear in the list in ascending numerical order.

### 7.20.1 Sense-key specific *changes*

*The following text should be added at the end of 7.20.1.*

If the sense key is COPY ABORTED and the SKSV bit is one, the SENSE-KEY SPECIFIC field shall be as shown in table 44.

**Table 44 — Segment pointer bytes**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 15 | SKSV | Reserved | SD | Reserved | BPV | BIT POINTER | | |
| 16 | (MSB) | FIELD POINTER | | | | | | |
| 17 | | | | | | | | (LSB) |

The segment descriptor (SD) bit indicates whether the field pointer is with reference to the start of the parameter list or to the start of a segment descriptor. An SD value of zero indicates that the field pointer is relative to the start of the parameter list. An SD value of one indicates that the field pointer is relative to the start of the segment descriptor indicated by the third and fourth bytes of the COMMAND SPECIFIC INFORMATION field (see 7.1.2).

A bit pointer valid (BPV) bit of zero indicates that the value in the BIT POINTER field is not valid. A BPV bit of one indicates that the BIT POINTER field specifies which bit of the byte designated by the FIELD POINTER field is in error. When a multiple-bit field is in error, the BIT POINTER field shall point to the most-significant (left-most) bit of the field.

The FIELD POINTER field indicates which byte of the parameter list or segment descriptor was in error.

NOTE 6 If the parameter list is in excess of 65528 bytes in length and SD is 0, the FIELD POINTER value may not fit in two bytes provided by the sense key specific format definition.

## 8 Other SPC-2 Changes

SPC-2 needs to be searched for references to the COPY command, and where applicable, the EXTENDED COPY command needs to be listed in those places where references to the COPY command are found. Examples of this condition at the time this proposal was written are as follows:

a) 3.1.13 - The definition of ‘copy manager’
b) 3.1.54 - The definition of ‘third-party’
c) Clause 6 - the device model for processor devices
d) 7.16.2 - Third-party release
e) 7.20 - REQUEST SENSE command (needs to show that the SEGMENT NUMBER field does not apply to EXTENDED COPY, since that data is in the COMMAND-SPECIFIC INFORMATION field)
f) 7.20 - REQUEST SENSE command - item c) in the list describing the INFORMATION field
g) 7.20 - REQUEST SENSE command - description of the COMMAND-SPECIFIC INFORMATION field
h) 7.20.4 - Sense key and sense code definitions - definition of COPY ABORTED
i) 7.21.2 Third-party reservation
j) 7.21.3 Superseding reservations - note 34

The EXTENDED COPY and RECEIVE COPY RESULTS commands need to be added to reservations conflicts table (Table 6 in rev 9), to the commands table (Table 8 in rev 9), to the commands for processor devices (Table 134 in rev 9) and to Annex C.