

Date: Dec. 28, 1995
To: X3T10 Committee
From: Gerry Houlder, Seagate Technology
Subj: Set Capacity Proposal

This proposal came about because some customers want to define the capacity reported by a disk drive as something less than its full capacity. The main reason for this is customers wanting drives from different vendors to have exactly the same number of logical blocks. This is desired for ease of replacement, especially in disk array applications.

Another reason is that drives are getting too large for some operating systems to handle. For example, many operating systems will only handle 8 GB of storage on a single device and several vendors are making drives with a larger capacity than this.

Rev. 0: Initial proposal to modify the Block Descriptor in MODE SELECT so that the Number of Blocks Field is expanded to 4 bytes.

Rev. 1: In discussions at the July 11 working group meeting, about 60% of that group felt that the backwards compatibility issues caused by extending the MODE SELECT Block Descriptor to 4 bytes made this option unacceptable. The group did favor adding a set capacity capability to the READ CAPACITY command. Rev. 1 related the changes needed to do this.

Rev. 2: In discussions at Sept. 12 working group meeting, the group requested that any reference to using the set capacity feature to split drive capacity across multiple LUNs be deleted. There were too many unresolved issues (such as how FORMAT command operated on the multiple LUNs) for the group to accept this feature. Also some wording changes were made in the paragraph describing reservation conflict cases.

Rev. 3: In discussions at Nov. 7 working group meeting, it was pointed out that several companies in the industry will use the technique described in Rev. 0 of this proposal. The group decided it was better to document that technique rather than create a new one that is radically different.

I propose using a new block descriptor format (with a 4 byte number of blocks field) for direct access, write-once, CD-ROM, optical, and array peripheral device types (codes 00h, 04h, 05h, 07h, and 0Ch respectively). These are the types that refer to the SBC document for at least some of their commands. All other types will use the existing block descriptor format. Some types (e.g., streaming devices) need the density code field to stay where it is.

This is better than using a "block descriptor type" field in the header because only the 10 byte MODE SELECT version of the header has room to add such a field. This would require the 6 byte version to stay with the existing format, which has a 3 byte number of blocks field.

The SPC must add a new mode parameter block descriptor section. It should read as described on the next page. If this proposal is accepted, the existing block descriptor wording can be changed to delete wording specific to SBC type devices since those devices will be covered by this new block descriptor. The document editor shall be empowered to make such related changes to the existing block descriptor wording that he deems appropriate.

[New section to be added to SPC document]

Table xx - Mode parameter block descriptor for block devices

Bit	7	6	5	4	3	2	1	0
Byte								
0	(MSB)							
1	Number of Blocks							
2								
3								
4	Density Code							
5	(MSB)							
6	Block Length							
7								
								(LSB)

This block descriptor format applies only to direct-access, write-once, optical memory, and array controller devices. All other device types use another block descriptor format. Block descriptors specify some of the medium characteristics for a logical unit. Support for block descriptors is optional. Each block descriptor contains a density code field, a number of blocks field, and a block length field. A unit attention condition as described in clause 7.8 shall be generated when any block descriptor values are changed.

The number of blocks field specifies the number of logical blocks on the medium to which the density code and block length fields apply. A value of zero indicates that all of the remaining logical blocks of the logical unit shall have the medium characteristics specified. If the number of blocks field contains 00FFFFFFh, it shall be interpreted to mean that the number of blocks is greater than 00FFFFFFEh. In this case the READ CAPACITY command may be used to determine the actual number of blocks for the device. (See SBC standard for READ CAPACITY command.)

If the SCSI device doesn't support changing its capacity by changing the number of blocks field (via a MODE SELECT command), the value in the number of blocks field is ignored. If the device supports changing its capacity by changing the number of blocks field, then the number of blocks field is interpreted as follows:

- a) If the number of blocks is set to zero, the device shall be set to its maximum capacity. If the block size has not changed, the device shall not become format corrupted. This capacity setting shall be retained through reset events or power cycles.
- b) If the number of blocks is greater than zero and less than or equal to its maximum capacity, the device shall be set to that number of blocks. If the block size has not changed, the device shall not become format corrupted. This capacity setting shall be retained through reset events or power cycles.
- c) If the number of blocks field is set to a value greater than the maximum capacity of the device then the command is terminated with a CHECK CONDITION status, the sense key is set to ILLEGAL REQUEST, and the additional sense code is set to LOGICAL BLOCK ADDRESS OUT OF RANGE. The device retains its previous block descriptor settings.

NOTES

52 There may be implicit association between parameters defined in the pages and block descriptor. For direct-access devices, the block length affects the optimum values (the value that achieves the best performance) for the sectors per track, bytes per physical sector, track skew factor, and cylinder skew factor fields in the format parameters page. In this case, the target may change parameters not explicitly sent with the MODE SELECT command. A subsequent MODE SENSE command would reflect these changes.

The density code field is unique for each device type. Refer to the mode parameters clause of the specific device type command standard (e.g., SBC) for the definition of this field. Some device types reserve all or part of this field.

The block length specifies the length in bytes of each logical block described by the block descriptor. If the block length is changed, the new value is not saved until a FORMAT command is received. When the block length is changed, the device shall indicate a format corrupted condition until a FORMAT command is received or the block length is restored to its previous value.