

CONGRUENT SOFTWARE, INC.
98 Colorado Avenue
Berkeley, CA 94707
(510) 527-3926
(510) 527-3856 FAX

FROM: Peter Johansson
TO: T10 SBP-3 working group
DATE: July 15, 2002
RE: Prototype AVD commands

This document describes a set of commands appropriate for devices that support both streaming audio/video and direct access to the device medium; such devices are referred to as AV direct-access (AVD) devices.

There is no intention that a separate device type be created for AVD devices. Instead, these commands are offered as prototypes for adoption into existing or future command sets. The adopting command set is responsible to assign opcodes for the commands.

Because most of the substantive information associated with each command is transport protocol-dependent, a relatively small set of commands suffices to provide the functionality needed:

- a) CONFIGURE STREAM parameterizes the stream with information such as start and stop times, how to identify the stream (from possibly many streams) both on the external transport medium (bus or other interconnect) and the device medium;
- b) PLAY and RECORD place the device in playback or recording mode, respectively. An associated "play list" describes the medium locations from which or to which the stream flows; and
- c) STOP interrupts any activity that may be in progress and unconditionally places the device in its idle mode while PAUSE and RESUME permit active playback or recording to be suspended and then restarted from the same device medium location.

The PLAY and RECORD commands differ from most existing SCSI commands in that they are modal. Unlike other commands which do not return completion status until the command has either succeeded or failed, AVD commands return completion status as soon as the requested operation is commenced—even though the device operation initiated may not complete until some future time. This is a key element of their design; it permits the device to remain ready to accept a new command even while it is active in a particular operational mode, such as playback or recording.

AV Direct-Access Commands

The AV Direct-Access (AVD) command set is shown in Table 1. The command set is intended to be supplementary to the principal command set implemented by the device logical unit.

Table 1 - AVD Command set

Command name	OpCode	Command Support
CONFIGURE STREAM	XXh	M
PLAY	XXh	O
RECORD	XXh	O
STOP	XXh	M
PAUSE	XXh	O
RESUME	XXh	O
Command Support key: M = support is mandatory; O = support is optional.		

Although the PLAY and RECORD commands are optional, the device logical unit shall implement at least one and may implement both.

The CONTROL byte (the last byte of the CDB) shall be set to zero.

CONFIGURE STREAM command

The CONFIGURE STREAM command (see Table 2) provides a means for the initiator to supply transport protocol-dependent information that characterizes the stream's representation on both the transport medium and device medium. This information may include, but is not limited to, synchronization times (*e.g.*, start or stop times), data format or information that identifies the stream data externally on the transport medium and internally on the device medium, (*e.g.*, a channel number).

Table 2 - CONFIGURE STREAM Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (XXh)							
1	(MSB)	RESERVED						(LSB)
3	(MSB)	ALLOCATION LENGTH						(LSB)
5	CONTROL = 00h							

The ALLOCATION LENGTH field specifies the size of the stream configuration information provided by the initiator. The format of stream configuration information is transport protocol-dependent and is specified either in a separate annex within this standard or in a separate document.

The device shall complete configuration as specified by the information provided by the initiator prior to the return of GOOD status for the command.

PLAY command

The PLAY command (see Table 3) provides a means for the initiator to request the device to playback stream data from its medium.

Table 3 - PLAY Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (XXh)							
1	(MSB)							
2	ALLOCATION LENGTH							
3								
4								
5	CONTROL = 00h							

The ALLOCATION LENGTH field specifies the size of the stream descriptor information provided by the initiator. The format of stream descriptor information is transport protocol-dependent and is specified either in a separate annex within this standard or in a separate document.

The return of GOOD status for the command indicates that the device has entered playback mode and has started to process the stream descriptor information. Although the command is complete (*i.e.*, the device is ready to accept additional commands), the device remains in playback mode until all the stream descriptor information has been processed or another command issued that changes the device's mode. No additional status is returned when the device completes processing the stream descriptor information.

RECORD command

The RECORD command (see Table 4) provides a means for the initiator to request the device to record stream data onto its medium.

Table 4 - RECORD Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (XXh)							
1	(MSB)							
2	ALLOCATION LENGTH							
3								
4								
5	CONTROL = 00h							

The ALLOCATION LENGTH field specifies the size of the stream descriptor information provided by the initiator. The format of stream descriptor information is transport protocol-dependent and is specified either in a separate annex within this standard or in a separate document.

The return of GOOD status for the command indicates that the device has entered recording mode and has started to process the stream descriptor information. Although the command is complete (*i.e.*, the device is ready to accept additional commands), the device remains in recording mode until all the stream descriptor information has been processed or another command issued that changes the device's mode. No additional status is returned when the device completes processing the stream descriptor information.

STOP command

The STOP command (see Table 5) provides a means for the initiator to place the device in its idle mode. If either the playback or recording mode is active, whether paused or not, all stream descriptor information associated with the mode shall be discarded. The STOP command may take effect immediately or may be synchronized with some event described by transport protocol-dependent information associated with the command.

Table 5 - STOP Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0	
0	OPERATION CODE (XXh)								
1	(MSB)								
...	TRANSPORT PROTOCOL-DEPENDENT								
8								(LSB)	
9	CONTROL = 00h								

The format of the TRANSPORT PROTOCOL-DEPENDENT field is specified either in a separate annex within this standard or in a separate document

The device shall cease the playback or recording of stream data and enter its idle mode prior to the return of GOOD status for the command.

PAUSE command

The PAUSE command (see Table 6) provides a means for the initiator suspend device playback or recording operations, if active. The PAUSE command may take effect immediately or may be synchronized with some event described by transport protocol-dependent information associated with the command.

Table 6 - PAUSE Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0	
0	OPERATION CODE (XXh)								
1	(MSB)								
...	TRANSPORT PROTOCOL-DEPENDENT								
8								(LSB)	
9	CONTROL = 00h								

The format of the TRANSPORT PROTOCOL-DEPENDENT field is specified either in a separate annex within this standard or in a separate document.

The device shall suspend the playback or recording of stream data prior to the return of GOOD status for the command.

RESUME command

The RESUME command (see Table 7) provides a means for the initiator resume device playback or recording operations, if paused. The RESUME command may take effect immediately or may be synchronized with some event described by transport protocol-dependent information associated with the command.

Table 7 - RESUME Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0	
0	OPERATION CODE (XXh)								
1	(MSB)								
...	TRANSPORT PROTOCOL-DEPENDENT								
8								(LSB)	
9	CONTROL = 00h								

The format of the TRANSPORT PROTOCOL-DEPENDENT field is specified either in a separate annex within this standard or in a separate document

The device shall resume the playback or recording of stream data prior to the return of GOOD status for the command.

Annex X (normative)

Serial Bus data structures for AVD

This annex specifies the transport protocol-dependent data structures referenced, but not defined by, the AVD command set.

X.1 Stream descriptor

A stream descriptor (sometimes called a "play list" in other contexts) is associated with an AVD PLAY or RECORD command; it provides the correlation between the stream as it appears on the external transport medium and the locations the stream occupies on the device medium. The stream descriptor may also contain other information that modifies device behavior as it plays back or records. A stream descriptor is an ordered set of one or more stream descriptor elements, whose format is illustrated by Figure 1.

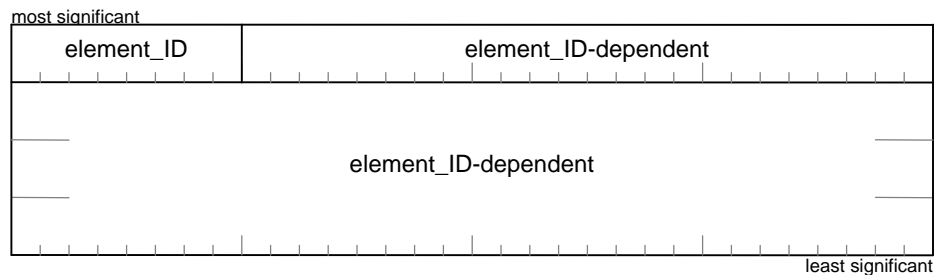


Figure 1 – Stream descriptor element

The *element_ID* field determines the format of the remainder of the stream descriptor element; permissible values are specified by the table below.

Value	Element ID
0	Segment descriptor
1	Stream configuration
2	STOP function
3	PAUSE function
4	RESUME function
5 – F ₁₆	Reserved for future standardization

The most common stream descriptor element is a segment descriptor, which has the format specified by Figure 2. The data described by a stream descriptor consists of one or more segments. Data is contiguous within a segment but segments need not be contiguous on the device medium with respect to each other.

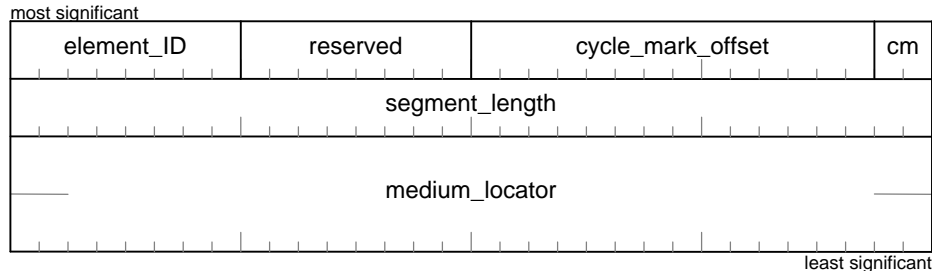


Figure 2 – Segment descriptor

The *cm* field (together with the *cycle_mark_offset* field) specifies the location of the first quadlet of isochronous data (segment offset) as encoded by the table below.

Value	<i>cycle_mark_offset</i>	Segment offset
0	Undefined	Zero
1	Undefined	<i>cycle_mark_offset</i>
2	Location of first CYCLE MARK	Zero
3	Location of first CYCLE MARK	<i>cycle_mark_offset</i>

The segment offset derived from the combination of *cm* and *cycle_mark_offset* specifies the location of the first quadlet of the isochronous data as an offset, in quadlets, relative to the starting medium location indicated by the *medium_locator*. For a block device, the segment offset, expressed in bytes, shall be less than the block size of the device.

NOTE – When an immediate segment descriptor is used, the command transported by the stream command block ORB specifies a starting location on the medium and an associated transfer length. Particularly in the case of block devices, the relevant isochronous data may be a subset of the data length and may commence at a nonzero offset relative to the natural block boundaries of the medium—hence the necessity for the additional values, *segment_length* and segment offset, to completely characterize the request.

The *cycle_mark_offset* field, when *cm* has a value of two or three, specifies the location of the first CYCLE MARK packet as an offset, in quadlets, relative to the starting medium location indicated by the *medium_locator*. When *cm* has a value of one, *cycle_mark_offset* specifies the segment offset instead. In either case, the value of *cycle_mark_offset*, converted to bytes, shall be less than *segment_length*.

NOTE – For a PLAY command, the *cycle_mark_offset* field may be useful to reestablish synchronization within the recorded isochronous data if a nonrecoverable error occurred while reading from the device medium.

The *segment_length* field specifies the length of data, in bytes, that is to be transferred to or from the device medium and shall be a multiple of four.

The *medium_locator* field shall specify the location on the device medium for the stream data. The meaning and usage of this field is determined by the device's command set. For block devices, the *medium_locator* usually contains a logical block address (LBA).

The other stream descriptor elements use a generic format that permits immediate information to be encoded within the four-quadlet element itself or in a data structure that immediately follows the

stream descriptor element. Except for segment descriptors, Stream descriptor elements have the format specified by Figure 3.

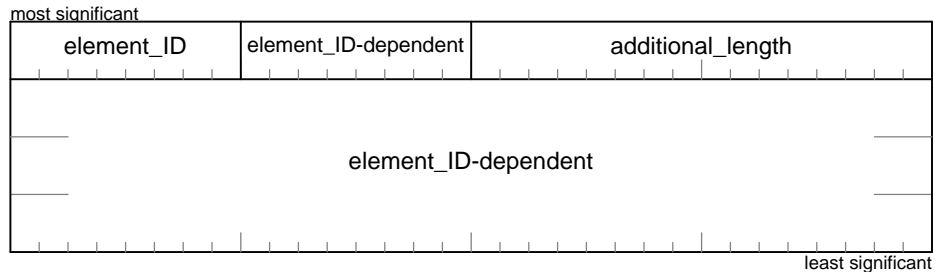


Figure 3 –Stream descriptor element (except segment descriptors)

The *additional_length* field shall specify the amount of information, in bytes, that follows the stream descriptor element; the value of the field shall be a multiple four.

When the additional information that follows a stream descriptor element contains stream configuration parameters, it shall have the format specified by Figure 4.

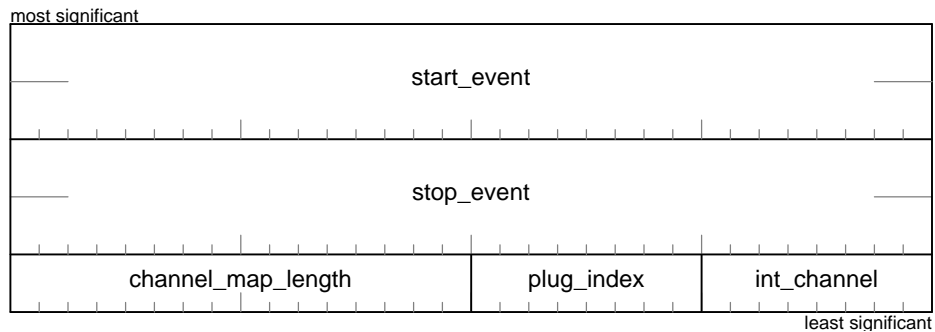


Figure 4 –Stream configuration information

The *start_event* and *stop_event* fields shall specify the point in time at which stream operations commence or stop, respectively. Both fields use the same format, illustrated by Figure 5.

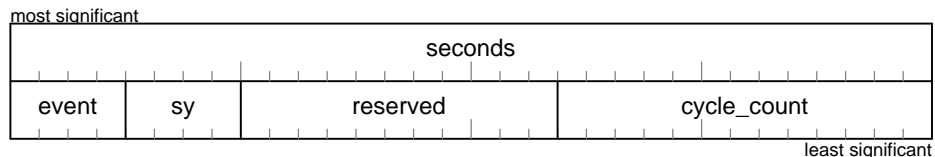


Figure 5 –Event descriptor

The *seconds* and *cycle_count* fields together identify a Serial Bus time that may be used for command synchronization when the *event* field specifies CYCLE MATCH. The time identified by these fields is compared with the device's cycle clock. An equal comparison occurs if the *seconds* field matches the device's BUS_TIME register and if the *cycle_count* field matches the field of the same name in the device's CYCLE_TIME register.

The *event* field, in combination with the *sy* field or the *seconds*, and *cycle_count* fields, identifies an externally observable Serial Bus event or time that may be used as a synchronization point for the action described by the associated command. Whether or not the command makes use of the event or time identified by these fields is not specified by this standard.

Value	Event code
0	NO EVENT SPECIFIED
1	IMMEDIATE
2	CYCLE MATCH
3	SY MATCH
4	FIRST DATA
5 – F ₁₆	Reserved for future standardization

A value of NO EVENT SPECIFIED indicates that the event is not meaningful for the command associated with the stream descriptor.

A value of IMMEDIATE instructs the device to commence the action indicated by the command as soon as possible.

A value of CYCLE MATCH instructs the device to commence the action indicated by the command at the cycle time specified by *seconds* and *cycle_count*.

A value of SY MATCH instructs the device to commence the action indicated by the command on the isochronous period for which the *sy* field of an isochronous packet for any enabled channel matches the *sy* field in the stream descriptor. A *event* value of SY MATCH is meaningful only for the *start_event* field and in combination with a RECORD command or if the device is already recording.

A value of FIRST DATA instructs the device to perform the specified action when isochronous data is observed for any enabled isochronous channel. A *event* value of FIRST DATA is meaningful only for the *start_event* field and in combination with a RECORD command or if the device is already recording.

NOTE – A *event* field value of FIRST DATA may have effects similar to IMMEDIATE, in that it is possible for isochronous data to be recorded immediately. The difference between the two stream events is apparent if no isochronous packets for any of the enabled channels are present when the command is executed. If IMMEDIATE is specified, CYCLE MARK packets are recorded as each cycle start is observed. If FIRST DATA is specified, no packets are recorded until the first isochronous packet for an enabled channel is observed. When this event occurs, a CYCLE MARK packet with the most recent cycle start data is recorded followed by a DATA packet(s) for the enabled channel(s).

The *sy* field is valid only for the *start_event* field and in combination with a RECORD command (or if the device is already recording) and the *event* field specifies SY MATCH. See the preceding description of *event*.

The *channel_map_length* field shall be zero when only one stream is to be transferred to or from the device medium; in this case, the stream data shall be identified by the *plug_index*[0] and *int_channel*[0] fields. When more than one stream is active, the *channel_map_length* field shall be a multiple of two and shall specify the size of the array of {*plug_index*, *int_channel*} pairs that follow.

The *plug_index* fields may uniquely identify a single source or sink or stream data within the context of the target. If *plug_index* is zero, command set-dependent methods specify the parameters of all stream data. When a *plug_index* is in the range one to 1F₁₆, inclusive, it identifies a plug control register that mediates reception or transmission of stream data. The type of plug control register referenced, either an input or output plug control register, is determined by the value of the *talker* bit.

Consult IEC 61883-1 for the specification of plug control registers. Other values of *plug_index* are reserved for future standardization.

The *int_channel* fields shall specify the channel number used internally by the device for each stream's isochronous packets. Some devices (or some of their operating modes) do not use internal channel numbers, in which case the field is ignored. For other devices, the combination of *int_channel[n]* and *plug_index[n]* specifies a transformation between internal and external (Serial Bus) channel numbers, dependent on whether the device is listening or talking. When the device is a listener, the observed channel number in the isochronous packet header shall be replaced with *int_channel* as the data is recorded on the medium. When the device is a talker, the channel number obtained from the medium shall, at the time the isochronous packet is transmitted, be replaced with the channel number implicitly identified by *plug_index*.

X.2 Immediate data (AVD STOP, PAUSE and RESUME commands)

The eight bytes of immediate, transport protocol-dependent data in the STOP, PAUSE and RESUME commands shall contain an event descriptor in the format specified by Figure 5. The device shall take the requested action at the synchronization time specified by the event descriptor.