Document number 2004005

AV/C Tape Recorder/Player Subunit Specification 2.4

September 1, 2004

Sponsored by:

1394 Trade Association

Accepted for publication by

1394 Trade Association Board of Directors

Abstract

This specification defines a command set for consumer and professional Tape recorder/player equipment over IEEE Std 1394-1995. The command set makes use of the Function Control Protocol (FCP) defined by IEC61883, proposed standard for Digital Interface for Consumer Electronic Audio/Video Equipment, for the transport of Tape recorder/player command requests and responses. The audio/video devices are implemented as a common unit architecture within IEEE Std 1394-1995.

Keywords

Audio, Video, 1394, Digital, Interface.

1394 Trade Association Specification

1394 Trade Association Specifications are developed within Working Groups of the 1394 Trade Association, a non-profit industry association devoted to the promotion of and growth of the market for IEEE 1394-compliant products. Participants in Working Groups serve voluntarily and without compensation from the Trade Association. Most participants represent member organizations of the 1394 Trade Association. The specifications developed within the working groups represent a consensus of the expertise represented by the participants.

Use of a 1394 Trade Association Specification is wholly voluntary. The existence of a 1394 Trade Association Specification is not meant to imply that there are not other ways to produce, test, measure, purchase, market or provide other goods and services related to the scope of the 1394 Trade Association Specification. Furthermore, the viewpoint expressed at the time a specification is accepted and issued is subject to change brought about through developments in the state of the art and comments received from users of the specification. Users are cautioned to check to determine that they have the latest revision of any 1394 Trade Association Specification.

Comments for revision of 1394 Trade Association Specifications are welcome from any interested party, regardless of membership affiliation with the 1394 Trade Association. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

Interpretations: Occasionally, questions may arise about the meaning of specifications in relationship to specific applications. When the need for interpretations is brought to the attention of the 1394 Trade Association, the Association will initiate action to prepare appropriate responses.

Comments on specifications and requests for interpretations should be addressed to:

Editor, 1394 Trade Association
1560 East Southlake Blvd., Suite 242
Southlake, TX 76092
USA

Published by

1394 Trade Association
1560 East Southlake Blvd., Suite 242
Southlake, TX 76092 USA

Printed in the United States of America

Contents

Tables

Figures

Annexes

Foreword (This foreword is not part of 1394 Trade Association Specification 2004005)

This specification defines a command set for consumer and professional Tape recorder/player equipment over IEEE Std 1394-1995. The command set makes use of the Function Control Protocol (FCP) defined by IEC61883, proposed standard for Digital Interface for Consumer Electronic Audio/Video Equipment, for the transport of Tape recorder/player command requests and responses. The audio/video devices are implemented as a common unit architecture within IEEE Std 1394-1995.

There is 1 normative annex in this specification. Annexes A, AV/C Tape recorder/player subunit commands in numerical order.

This specification was accepted by the Board of Directors of the 1394 Trade Association. Board of Directors acceptance of this specification does not necessarily imply that all board members voted for acceptance. At the time it accepted this specification, the 1394 Trade Association. Board of Directors had the following members:

Eric Anderson, Chair
Angela Lee, Vice-Chair
Dave Thompson, Secretary

Organization Represented	*Name of Representative*
Apple Computer	Eric Anderson
Agere Systems	Dave Thompson
Firecomms Ltd.	Declan O'Mahney
Hewlett-Packard	Alan Berkerma
Matsushita/Panasonic	Hans van der Ven
Microsoft Corporation	Steve Powers
Mindready Solutions	Martin Lessard
Mitsubishi Digital Electronics America, Inc.	Angela Lee
Molex	Max Bassler
Newnex Technologies	Sam Liu
Oxford Semiconductor	Jalil Oraee
Quantum Parametrics LLC	Richard Mourn
Samsung Electronics	Jong-Wook Park
Staccato Communications	Jason Ellis
Texas Instruments	Cecelia Smith
VividLogic	Shiva Patibanda

The AV Working Group, which developed and reviewed this specification, had the following members:

Tsuyoshi Sawada, Chair
Takuya Nishimura, Secretary

Ram Balareaman
Enriqve Bastante
Hiroyuki Chaki
Jun-ichi Fujimori
Jim Haagen-Smit
Don Harwood
Burke Henehan
Hiroyuki Iitsuka
Prashant Kanher
Mike Kent
Jong Won Kim
Pascal Lagrange
Morten Lave
Matt Majid
Phil Maness
Matt Mora
Richard Mourn
Yuji Nakura
Yasutoshi Onishi
Takeshi Okauchi
Jalil Oraee
Mike Overlin
Jongwook Park
Song Hee Park
Shiva Patibanda
Alan Perry
Jehu Pineda
Tommy Shih
Shigeki Takahashi
Michael Johas Teener
Thomas Thaler
Nick Thompson
Niel Warren
Colin Whitby-strevens
Michael Yang
Andy Yanowitz
Jun-ichi Yoshio

Revision history

Changes from previous version

Version 2.4 differs from version 2.3 in the following:

- Definitions of signal mode value for HDV were added in

  4.9 INPUT SIGNAL MODE Command

  4.14 OUTPUT SIGNAL MODE Command

Version 2.3 differs from version 2.2 in the following:

- Definition of a signal_mode value for audio tape (Table 21).
- Definition of cassette_type and tape_grade_and_write_protect values for audio tape (Table 24).

Version 2.2 differs from version 2.1 in the following:

- According to the cassette name change, Neo was changed to MICROMV.
- Enhancement to the AV/C Tape Recorder/Player Subunit Specification Version 2.1 was merged.
- Enhancements for MICROMV cassette were added in

  4.3 ABSOLUTE TRACK NUMBER Command

  4.9 INPUT SIGNAL MODE Command

  4.12 MEDIUM INFO Command

  4.20 RECORDING SPEED Command

Version 2.1 differs from version 2.0.1 in the following:

- Renamed VCR subunit to Tape recorder/player subunit
- 8mm support was added
- Enhancements for D-VHS were added
- Some minor error corrections were made

Table 1 – Additional or/and Modified Commands

Item	Additional or/and Modified Command	Reference
8mm support	AREA MODE command	4.2
	ABSOLUTE TRACK NUMBER command	4.3
	BACKWARD command	4.5
	EDIT MODE command	4.7
	INPUT SIGNAL MODE command	4.9
	MEDIUM INFO command	4.12
	TIME CODE command	4.28
D-VHS digital & analog recording speed	RECORDING SPEED	4.20
Format selection for D-VHS	TAPE PLAYBACK FORMAT	4.26
	TAPE RECORDING FORMAT	4.27

Version 2.0.1 differs from version 2.0 in the following ways:

– The AV/C Digital Interface Command Set 2.0 manual was separated into two books: General Specification and the VCR Subunit Specification

Version 2.0 differs from version 1.0 in the following ways:

– Error corrections from the previous version: the opcode for RECORDING TIME was corrected in 4.21 to be 0x54

– The WRITE MIC command description was enhanced to allow the same response method for INTERIM as for ACCEPTED.

– The following commands are newly defined (either for the D-VHS model, or as enhancements to the AV/C VCR command set):

Table 2 – New commands in Version 2.0

AREA MODE
BINARY GROUP
MARKER
RELATIVE TIME COUNTER
SMPTE/EBU RECORDING TIME
SMPTE/EBU TIME CODE
TAPE PLAYBACK FORMAT
TAPE RECORDING FORMAT

– The following (existing) commands have been modified to support the newly defined D-VHS VCR model:

Table 3 – Modified commands in Version 2.0

ABSOLUTE TRACK NUMBER
BACKWARD
EDIT MODE
FORWARD
INPUT SIGNAL MODE
MEDIUM INFO
OUTPUT SIGNAL MODE
PRESET
RECORD
RECORDING DATE
RECORDING SPEED
RECORDING TIME
SEARCH MODE
TIME CODE

AV/C Tape Recorder/Player Subunit Specification 2.4

1 Preface

This document describes a model and command set for AV/C Tape recorder/player Subunits. For a complete overview of the AV/C specification, please refer to the AV/C Digital Interface Command Set General Specification, which is in a separate document. The suite of AV/C documentation is separated into a general AV/C specification document and separate documents for each type of subunit (Tape recorder/player, Tuner, Disc, etc.).

2 Normative references

2.1 Reference scope

The specifications and standards named in this section contain provisions, which, through reference in this text, constitute provisions of this 1394 Trade Association Specification. At the time of publication, the editions indicated were valid. All specifications and standards are subject to revision; parties to agreements based on this 1394 Trade Association Specification are encouraged to investigate the possibility of applying the most recent editions of the specifications and standards indicated below.

2.2 Approved references

The following approved specifications and standards may be obtained from the organizations that control them.

[1] IEEE Std 1394-1995, Standard for a High Performance Serial Bus

[2] IEEE Std 1394a-2000, Standard for a High Performance Serial Bus—Amendment 1

[3] IEEE Std 1394b-2002, Standard for a High Performance Serial Bus—Amendment 2

[4] AV/C Digital Interface Command Set General Specification, version 2.0.1, January 5, 1998

[5] IEC 61883, Consumer Electronic Audio/Video Equipment Digital Interface (1998-02)

[6] ISO/IEC 13123:1994, Control and Status Register (CSR) Architecture for Microcomputer Buses

[7] HD Digital VCR Conference, Specifications of Consumer-Use Digital VCR's using 6.3 mm magnetic tape (December 1995)

[8] IEC 61834, Recording – Helical-scan digital video cassette recording system using 6,35 mm magnetic tape for consumer use (1998-08)

[9] 8mm Video Conference c/o Electronic Industries Association of Japan, HELICAL-SCAN VIDEO TAPE CASSETTE SYSTEM USING 8mm MAGNETIC TAPE KNOWN AS 8mm VIDEO (April 1984)

[10] ANSI/SMPTE 12M-1995 (Revision of ANSI/SMPTE 12M-1986), SMPTE Standard for Television, Audio and Film - Time and Control Code.

[11] ANSI/SMPTE 262M-1995, SMPTE Standard for Television, Audio and Film - Binary Groups of Time and Control Codes - Storage and Transmission of Data.

[12] Victor Company of Japan, Limited (JVC), D-VHS SYSTEM STANDARD

[13] Victor Company of Japan, Limited (JVC), VHS VIDEO CASSETTE SYSTEM STANDARD

[14] Victor Company of Japan, Limited (JVC), S-VHS VIDEO CASSETTE SYSTEM STANDARD

[15] Specification of HDV Recording Format Version 1.0, September 2003 (http://www.hdv-info.org/)

Throughout this document, the term "IEEE 1394" shall be understood to refer to IEEE Std 1394-1995 as amended by IEEE Std 1394a-2000 and IEEE Std 1394b-2002.

2.3 References under development

At the time of publication, the following referenced specifications and standards were under development.

2.4 Reference acquisition

The references cited may be obtained from the organizations that control them:

1394 Trade Association, 1560 East Southlake Blvd., Suite 242, Southlake, TX 76092 USA; (817) 416-2200 / (817) 416-2256 (FAX); http://www.1394ta.org/

American National Standards Institute (ANSI), 25 West 43rd Street, 4 floor, New York, NY 10036, USA; (212) 642-4900 / (212) 398-0023 (FAX); http://www.ansi.org/

Institute of Electrical and Electronic Engineers (IEEE), 445 Hoes Lane, PO Box 1331, Piscataway, NJ 08855-1331, USA; (732) 981-0060 / (732) 981-1721 (FAX); http://www.ieee.org/

In addition, many of the documents controlled by the above organizations may also be ordered through a third party:

Global Engineering Documents, 15 Inverness Way, Englewood, CO 80112-5776; (800) 624-3974 / (303) 792-2192; http://www.global.ihs.com/

3 Definitions and notation

3.1 Definitions

3.1.1 Conformance

Several keywords are used to differentiate levels of requirements and optionality, as follows:

3.1.1.1 expected: A keyword used to describe the behavior of the hardware or software in the design models assumed by this specification. Other hardware and software design models may also be implemented.

3.1.1.2 ignored: A keyword that describes bits, bytes, quadlets, octlets or fields whose values are not checked by the recipient.

3.1.1.3 may: A keyword that indicates flexibility of choice with no implied preference.

3.1.1.4 reserved: A keyword used to describe objects (bits, bytes, quadlets, octlets and fields) or the code values assigned to these objects in cases where either the object or the code value is set aside for future standardization. Usage and interpretation may be specified by future extensions to this or other specifications. A reserved object shall be zeroed or, upon development of a future specification, set to a value specified by such a specification. The recipient of a reserved object shall ignore its value. The recipient of an object defined by this specification as other than reserved shall inspect its value and reject reserved code values.

3.1.1.5 shall: A keyword that indicates a mandatory requirement. Designers are required to implement all such mandatory requirements to assure interoperability with other products conforming to this specification.

3.1.1.6 should: A keyword that denotes flexibility of choice with a strongly preferred alternative. Equivalent to the phrase “is recommended.”

3.1.2 Glossary

The following terms are used in this specification:

3.1.2.1 Operation Modes: Most of the AV/C Tape recorder/player commands can be executed in different operation modes. These modes are essentially variants of the main command, and affect what actually gets done with the data that is specified in the command. The mode parameter is normally operand[0] in the command frame. The following table clarifies the existing mode definitions, and introduces two new ones, called *unreliable* and *preset*:

Table 4 – Operation modes

mode name	mode value	meaning
search	20_{16}	Execute the command by searching the medium to find the specified location. The location may be specified by consumer or professional time code values.
Unreliable	61_{16}	Rather than REJECTing a command, some subunits may make a best guess attempt to return certain kinds of status information. When a subunit does this, it shall set the mode parameter in the response frame to 61_{16}, which means that the data has been "estimated."
Preset	70_{16}	When the command is executed, data on the medium is not changed. The preset mode allows the controller to establish or read preset values. These values will take effect (cause a change to data on the medium) when a command is executed some time in the future.
Data	71_{16}	When the command is executed, then data actually changes or is read from the medium itself.

In the text that describes the usage of a command, it is sometimes useful to indicate which operation mode is being referred to. This is done by writing the text as "COMMAND NAME, operation mode." For example, "BINARY GROUP, preset" indicates that the text is discussing the preset operation mode of the BINARY GROUP command.

Not all commands have these operation modes, and not all types of Tape recorder/player subunits support all operation modes of a particular command.

3.1.2.2 Consumer Time Code: The Consumer Time Code is defined as HH:MM:SS:FF (hours:minutes:seconds:frames). The following diagram illustrates the consumer time code format:

msb							lsb
frames (high nibble)				frames (low nibble)			
seconds (high nibble)				seconds (low nibble)			
minutes (high nibble)				minutes (low nibble)			
hours (high nibble)				hours (low nibble)			

Figure 1 – Consumer Time Code format

Each of the *frames*, *seconds*, *minutes* and *hours* fields is encoded in binary coded decimal (BCD) format.

3.1.2.3 SMPET/EBU Time Code: Please refer to section 4.25 SMPTE/EBU TIME CODE Command.

3.1.2.4 DVCR: Digital video cassette recording system as defined by IEC 61834 [8]

3.1.2.5 D-VHS: Digital video cassette recording system defined by D-VHS STANDARD [12]

3.1.2.6 VHS: Video cassette recording system as defined by VHS STANDARD [13]

3.1.2.7 S-VHS: Video cassette recording system as defined by S-VHS STANDARD [14]

3.1.2.8 8mm: Video cassette recording system as defined by 8mm Specification [9]

3.1.2.9 HDV: Digital video cassette recording system as defined by HDV Specification [15]

3.1.3 Abbreviations

The following are abbreviations that are used in this specification:

AV/C Audio Video Control

lsb least significant bit

msb most significant bit

3.2 Notation

3.2.1 Numeric values

Decimal and hexadecimal are used within this specification. By editorial convention, decimal numbers are most frequently used to represent quantities or counts. Addresses are uniformly represented by hexadecimal numbers. Hexadecimal numbers are also used when the value represented has an underlying structure that is more apparent in a hexadecimal format than in a decimal format.

Decimal numbers are represented by Arabic numerals without subscripts or by their English names. Hexadecimal numbers are represented by digits from the character set 0 – 9 and A – F followed by the subscript 16. When the subscript is unnecessary to disambiguate the base of the number it may be omitted. For the sake of legibility hexadecimal numbers are separated into groups of four digits separated by spaces.

As an example, 42 and $2A_{16}$ both represent the same numeric value.

3.2.2 Bit, byte and quadlet ordering

This specification uses the facilities of Serial Bus, IEEE 1394, and therefore uses the ordering conventions of Serial Bus in the representation of data structures. In order to promote interoperability with memory buses that may have different ordering conventions, this specification defines the order and significance of bits within bytes, bytes within quadlets and quadlets within octlets in terms of their relative position and not their physically addressed position.

Within a byte, the most significant bit, *msb*, is that which is transmitted first and the least significant bit, *lsb*, is that which is transmitted last on Serial Bus, as illustrated below. The significance of the interior bits uniformly decreases in progression from *msb* to *lsb*.

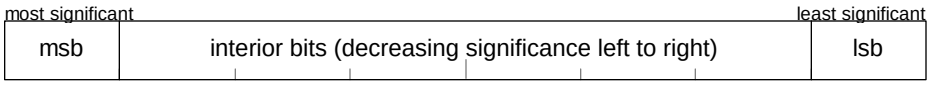

Figure 2 – Bit ordering within a byte

Within a quadlet, the most significant byte is that which is transmitted first and the least significant byte is that which is transmitted last on Serial Bus, as shown below.

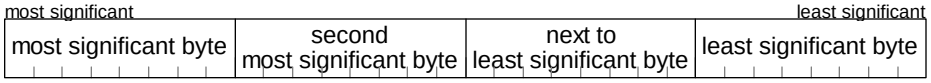

Figure 3 – Byte ordering within a quadlet

Within an octlet, which is frequently used to contain 64-bit Serial Bus addresses, the most significant quadlet is that which is transmitted first and the least significant quadlet is that which is transmitted last on Serial Bus, as the figure below indicates.

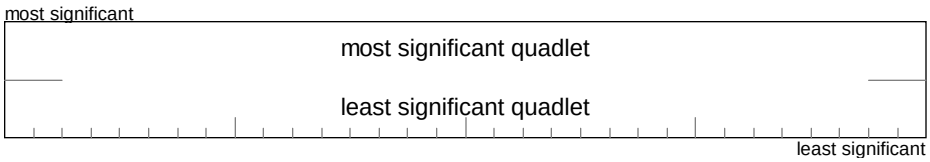

Figure 4 – Quadlet ordering within an octlet

When block transfers take place that are not quadlet aligned or not an integral number of quadlets, no assumptions can be made about the ordering (significance within a quadlet) of bytes at the unaligned beginning or fractional quadlet end of such a block transfer, unless an application has knowledge (outside of the scope of this specification) of the ordering conventions of the other bus.

4 Tape Recorder/Player Subunit Commands

Tape recorder/player subunit commands are identified by a *subunit_type* value of four and a *subunit_ID* value between zero and seven, inclusive. Table 5 below summarizes the Tape recorder/player subunit commands.

Table 5 – Tape recorder/player subunit commands

Opcode	Value	Support level (by *ctype*) C	S	N	Comments
ANALOG AUDIO OUTPUT MODE	70_{16}	O	O	–	Control analog audio signal
AREA MODE	72_{16}	O	O	–	Control command specifies the area on the medium for input signal(s) to be recorded
ABSOLUTE TRACK NUMBER	52_{16}	*	*	–	Report tape position
AUDIO MODE	71_{16}	O	O	–	Control audio signal recording mode
BACKWARD	56_{16}	R	–	–	Search for a tape position
BINARY GROUP	$5A_{16}$	O	O	O	Reads or writes the binary group preset data, reads the binary group data from the medium.
EDIT MODE	40_{16}	O	O	–	Control editing operations prior to an anticipated playback or record command
FORWARD	55_{16}	R	–	–	Search for a tape position
INPUT SIGNAL MODE	79_{16}	O	M	O	Control input signal mode
LOAD MEDIUM	$C1_{16}$	O	–	–	Control eject, open and close
MARKER	CA_{16}	R	R	O	Record or erase a marker signal
MEDIUM INFO	DA_{16}	–	R	–	Report medium information
OPEN MIC	60_{16}	*	R	–	Open or close MIC
OUTPUT SIGNAL MODE	78_{16}	O	M	O	Control output signal mode
PLAY	$C3_{16}$	*	–	–	Control the playback mode of the transport mechanism
PRESET	45_{16}	O	O	–	Establish operating parameters for the transport mechanism
READ MIC	61_{16}	R	–	–	Read data from MIC
RECORD	$C2_{16}$	*	–	–	Control the recording mode of the transport mechanism
RECORDING DATE	53_{16}	O	O	–	Report recording date
RECORDING SPEED	DB_{16}	O	O	–	Control recording speed
RECORDING TIME	54_{16}	–	O	–	Report recording time
RELATIVE TIME COUNTER	57_{16}	R	R	–	Search, inquire or clear the RTC value
SEARCH MODE	50_{16}	–	R	O	Report transport mechanism search mode status
SMPTE/EBU RECORDING TIME	$5C_{16}$	O	O	O	Reads or writes the preset recording time, reads the recording time from the medium. Uses the SMPTE/EBU time code format

Opcode	Value	Support level (by *ctype*) C	S	N	Comments
SMPTE/EBU TIME CODE	59_{16}	O	O	O	Reads or writes the preset time code, reads the time code from the medium. Uses the SMPTE/EBU time code format
TAPE PLAYBACK FORMAT	$D3_{16}$	*	*	–	Specifies the digital playback format
TAPE RECORDING FORMAT	$D2_{16}$	*	*	–	Specifies the digital record format
TIME CODE	51_{16}	*	*	–	Search or inquire about specified medium location
TRANSPORT STATE	$D0_{16}$	–	M	O	Report current state of transport mechanism
WIND	$C4_{16}$	*	–	–	Control transport mechanism motion when not in playback or record
WRITE MIC	62_{16}	O	O	–	Store data in MIC

In the preceding tables, a dash in the support level column indicates that the command is not defined for the *ctype* value, CONTROL, STATUS or NOTIFY, indicated. An asterisk in the support level column indicates that the command operands or the type of subunit determine whether the command is mandatory (M), recommended (R) or optional (O). The specific command formats and corresponding response frame formats are described for each of the commands in the clauses that follow.

4.1 ANALOG AUDIO OUTPUT MODE Command

The ANALOG AUDIO OUTPUT MODE control command configures a Tape recorder/player subunit to output an analog audio signal on its source plugs. The structure of the command is shown in Figure 5 below:

	Msb							lsb
opcode	ANALOG AUDIO OUTPUT MODE (70_{16})							
operand[0]	analog_audio_mode							

Figure 5 – ANALOG AUDIO OUTPUT MODE control command format

The *analog_audio_mode* field specifies the signal format of the analog audio output, as defined by the table that follows:

Table 6 – Analog audio mode

Value	Analog audio mode
20_{16}	Monaural
21_{16}	Right
22_{16}	Left
23_{16}	Stereo
25_{16}	Secondary
26_{16}	Main
27_{16}	Bilingual

In addition to the use of ANALOG AUDIO OUTPUT MODE as a control command, it may also be used to determine the current audio output signal mode available at the Tape recorder/player subunit's source plugs. The format used when *ctype* has a value of STATUS is illustrated by Figure 5.

	msb						lsb
opcode	ANALOG AUDIO OUTPUT MODE (70_{16})						
operand[0]	FF_{16}						

Figure 6 – ANALOG AUDIO OUTPUT MODE status command format

In the response frame returned by the Tape recorder/player subunit, *operand[0]* is replaced with the current analog audio output signal format as defined above.

4.2 AREA MODE Command

The AREA MODE control command specifies the area on the medium for input signal(s) to be recorded. It is used to prepare for an editing operation, which will later be carried out by issuing the EDIT MODE command. The AREA MODE command allows independent access to any part of an AREA. The format of the AREA MODE control command is illustrated by the figure below:

	msb						lsb
opcode	AREA MODE (72_{16})						
operand[0]	medium_type				area_number		
operand[1]	part[0]		part[1]		part[2]		part[3]
operand[2]	part[4]		part[5]		part[6]		part[7]
operand[3]	FF_{16}						
operand[4]	FF_{16}						

Figure 7 – AREA MODE control command format

The *medium_type* operand indicates the kind of the medium used in the Tape recorder/player subunit, as defined by Table 11.

The *area_number* operand identifies each of eight possible *rec_areas*, which are numbered between zero and seven inclusive, and represents the corresponding *rec_area* names AREA 0 through AREA 7 respectively, as defined by the Table 8.

The *part[n]* operands control whether or not a particular divided part of the *rec_area* is active, according to the values summarized below. Some kinds of recording areas may be divided into eight possible parts, and may be represented in the AREA MODE operands by an index, n, which will vary between zero and seven inclusive. These subdivided parts are represented by the names *part[0]* to *part[7]*, respectively.

The "don't care" value in any of the AREA MODE operands permits the Tape recorder/player subunit to select any permissible value for the operand.

Table 7 – AREA MODE operands

part[n]	Description
00_2	Record
01_2	Don't record
10_2	Reserved
11_2	Don't care

Table 8 – Area number assignment

area_number	rec_area	DVCR SD	D-VHS	8mm
0	AREA 0	ITI sector	Control track	not defined
1	AREA 1	Audio sector	Linear audio track	PCM audio area
2	AREA 2	Video sector	Main code area	Video (+AFM) area
3	AREA 3	Subcode sector	Subcode area	Subcode area
4	AREA 4	not defined	not defined	not defined
5	AREA 5	not defined	not defined	not defined
6	AREA 6	not defined	not defined	not defined
7	AREA 7	not defined	not defined	not defined

The actions performed by the AREA MODE command are dependent on the medium format, and the AREA being manipulated. For DVCR SD, this command is currently defined only for use in AREA 1 (audio insert recording of up to 4 channels). Operations on all other areas of SD format medium are not supported.

AREA 1 of the SD format is divided into two audio channels, CH1 and CH2. Each of these two channels can be further sub-divided into two audio channels, resulting in CHa, CHb, CHc and CHd. For recording only in CH1 or CH2, it is sufficient to use the AUDIO MODE command. To record in CHa through CHd, the AREA MODE command must be used. For the SD format, recording in CHa through CHd is only supported for the 32 kHz sampling rate.

In order to record in CHa through CHd, AREA MODE must be used in conjunction with the AUDIO MODE command. AUDIO MODE is used first to prepare the subunit for the kind of audio editing to be performed. When issuing the AUDIO MODE command to prepare for AREA MODE with SD format, only audio blocks CH1 and CH2 are considered; the operands *audio_ctrl*[2] and *audio_ctrl*[3] shall be set to the value 1 (don't record), and the operands *audio_channels*[2] and *audio_channels*[3] shall be set to 3 (don't care). The operands *sample_freq*[2] and *sample_freq*[3] shall be set to 3 (don't care).

To access CHa through CHd of the SD format, it is necessary to set *audio_ctrl*[0] and *audio_ctrl*[1] of AUDIO MODE to the value 0 (record), and their corresponding *audio_channels*[0] and *audio_channels*[1] parameters to the value 1 (use two audio channels per audio block). Specifying two channels for each audio block will result in a total of 4 audio channels being available for recording. The *sample_freq*[0] and *sample_freq*[1] fields shall be set to 2, indicating 32 kHz sample rate. Specifying any other parameters than those described here will ultimately result in the AREA MODE command being REJECTED if a 4 channel audio insert operation is performed.

The following diagram illustrates the first 4 bytes of the command frame for AUDIO MODE that must be used in preparation for 4 channel recording with the AREA MODE command:

	msb							lsb
opcode	0	1	1	1	0	0	0	1
operand[0]	0	1	0	1	0	0	0	0
operand[1]	0	1	1	0	0	1	1	0
operand[2]	1	1	1	1	1	1	1	1

Figure 8 – AREA MODE command example (first 4 bytes)

It is possible to perform audio insert editing on individual channels CHa through CHd, by specifying a value of 0 (record) or 1 (don't record) for each of the part[0] through part[3] operands of the AREA MODE command. However, there is a relationship between the record/don't record bits of the AUDIO MODE and AREA MODE commands, which will directly affect the execution results of AREA MODE. The following table illustrates this relationship. For example, only one audio block is shown, but these rules apply equally to all audio blocks:

Table 9 – Relationship between AUDIO MODE and AREA MODE

AUDIO MODE	AREA MODE	RESULTS
If the audio block record bits are set to this value...	**...and the audio channel record bits are set to this value...**	**...then this will be the result when AREA MODE is issued**
CH1 = don't record	CHa or CHb = don't record	OK
CH1 = don't record	CHa or CHb = record	REJECTED
CH1 = record	CHa or CHb = don't record	OK
CH1 = record	CHa or CHb = record	OK

To summarize, if the AUDIO MODE record bits have been set to "don't record" for an audio block, and the AREA MODE record bits have been set to "record" for an audio channel, then the AREA MODE command shall be REJECTED. All other combinations are legal. The results shown in the table above assume that all other conditions are satisfied to permit the command to execute.

It is possible to issue the AUDIO MODE command only once to set the sample rate to 32 kHz and enable recording on CH1 and CH2. Then, many AREA MODE commands may be issued, each independently affecting the recording (or not recording) of CHa, CHb, CHc and CHd.

The AREA MODE command may also be used with a *ctype* value of STATUS. The format of this status command is illustrated by the figure below.

	msb							lsb
opcode	AREA MODE (72_{16})							
operand[0]	FF_{16}							
…								
operand[4]								

Figure 9 – AREA MODE status command format

The AV/C response frame returns the current status of the Tape recorder/player subunit in the operands, in the format already described in the preceding tables.

4.3 ABSOLUTE TRACK NUMBER Command

The ABSOLUTE TRACK NUMBER command has two functions determined by the value of *ctype* and *operand[0]*. The first requests the Tape recorder/player subunit to search for a specified absolute track number on the medium while the second requests the Tape recorder/player subunit to return the absolute track number value for the current medium position. The support levels for this command will depend on the type of Tape recorder/player subunit, as defined below:

Table 10 – ABSOLUTE TRACK NUMBER command support level

Type of Tape recorder/player subunit	Support level (by *ctype*)			Comments
	C	S	N	
DVCR	R	M	–	
D-VHS	O	O	–	
8mm	–	–	–	
MICROMV	R	M	–	

The first format of the ABSOLUTE TRACK NUMBER command shall have a *ctype* value of CONTROL and is illustrated by Figure 10 below:

	msb							lsb
opcode	ABSOLUTE TRACK NUMBER (52_{16})							
operand[0]	20_{16}							
operand[1]	ATN_data							
operand[2]								
operand[3]								
operand[4]	medium_type						medium_type_dependent	

Figure 10 – ABSOLUTE TRACK NUMBER control command format

For both DVCR and D-VHS, the ABSOLUTE TRACK NUMBER control command requests the Tape recorder/player subunit to be paused in playback mode immediately after searching for the specified absolute track number on the medium.

The *ATN_data* field contains either the *ATN_data* format defined as a 23-bit *absolute_track_number* and a 1-bit *blank flag(bf)* for DVCR, or the *ATN_data* format defined as a 22-bit *absolute_track_number* and a 2-bit *support flag(sf)* for D-VHS . These are illustrated below:

	msb							lsb
operand[1]	(least significant bits)							bf
operand[2]	absolute_track_number							
operand[3]	(most significant bits)							

Figure 11 – *ATN_data* format for DVCR

	msb							lsb
operand[1]	sf	(most significant bits)						
operand[2]	absolute_track_number							
operand[3]	(least significant bits)							

Figure 12 – *ATN_data* format for D-VHS VCR

	msb							lsb
operand[1]	(least significant bits)							
operand[2]	absolute_track_number							
operand[3]	bf	(most significant bits)						

Figure 13 – *ATN_data* for MICROMV

The *medium_type* operand indicates the kind of the medium used in the Tape recorder/player subunit; a *medium_type* value of 00001_2 represents the medium used in the D-VHS VCR while a *medium_type* value of 11111_2 represents the medium used in the DVCR, as defined by the table below:

Table 11 – Medium_Type value

medium_type	meaning	medium_type_dependent	meaning
00001_2	D-VHS	000_2 ... 110_2	reserved
		111_2	No information
00010_2	MICRO MV	000_2 ... 110_2	reserved
		111_2	No information
11111_2	DVCR	000_2 | 110_2	reserved
		111_2	No information

The *medium_type_dependent* operand indicates additional information for the medium specified by *medium_type*. A *medium_type_dependent* value of 111_2 is defined to indicate that no additional information is available.

The response frame format is identical to the control command frame.

The search is successful if a medium location is found such that the recorded values for *ATN_data* match the search criteria. See the discussion of the contents included in the *ATN_data* below for a more detailed explanation.

The second format of the ABSOLUTE TRACK NUMBER command shall have a ctype value of STATUS and is illustrated by Figure 14 below:

	msb							lsb
opcode	ABSOLUTE TRACK NUMBER (52_{16})							
operand[0]	71_{16}							
operand[1]	FF_{16}							
…								
operand[4]								

Figure 14 – ABSOLUTE TRACK NUMBER status command format

If the Tape recorder/player subunit is able to return a STABLE response to the ABSOLUTE TRACK NUMBER status command, the AV/C response frame has the format illustrated by Figure 15 below:

	msb							lsb
opcode	ABSOLUTE TRACK NUMBER (52_{16})							
operand[0]	71_{16}							
operand[1]	ATN_data							
operand[2]								
operand[3]								
operand[4]	medium_type						medium_type_dependent	

Figure 15 – ABSOLUTE TRACK NUMBER response format

In the case of DVCR, the *ATN_data* operand shall contain the 23-bit absolute track number and *bf* bit. The *bf* bit provides discontinuity information for the absolute track numbers recorded on the medium. At the time that medium is recorded, a Tape recorder/player subunit writes *bf* values of zero if either a) the present medium position does not have an absolute track number (unrecorded medium exists prior to the current location) or b) there is at least one recorded track with a zero *bf* prior to the current location. In cases where no recording discontinuity exists between the beginning of medium and the location specified by *absolute_track_number*, the *bf* value for all tracks shall be one. The *absolute_track_number* returned is encoded as specified by the HD Digital VCR Conference.

In the case of D-VHS, the *ATN_data* operand shall contain a 22-bit *absolute_track_number* and a 2-bit *sf*. The *absolute_track_number* is counted up at every track and recorded. The system to count *absolute_track_number* is defined using the following formula:

ATN = 2 x n

where

ATN: *absolute_track_number*

n = 0, 1, 2, 3, ...(the positive integers from 0, inclusive)

Also, the starting position of *absolute_track_number*(000000_{16}) shall be located on a position within thirty seconds from the beginning of the medium without regard to the program. For pre-recorded (commercial) medium however, the starting position of *absolute_track_number* may be located at the starting position of the program. This may be more than thirty seconds from the beginning of the medium.

The *sf* operand specifies the *absolute_track_number support flag* on the medium, as defined by the table below:

Table 12 – Absolute_Track_number support flag

Value	Description
00_2	At the present position of the medium, absolute track number is not supported.
01_2	The present absolute track number is the estimated value and it may not be unique.
10_2	The present absolute track number is the estimated value and it is unique, but there may exist discontinuity.
11_2	The present absolute track number is continuous from the beginning of the medium.

In case the recording of *absolute track number* begins in the midst of the medium, an estimated value by the Tape recorder/player subunit may be recorded. In that case, the *sf* shall be 01_2 within the double of the error(the difference between the estimated value and the value to be recorded if the *absolute_track_number* is kept recording continuously from the beginning). After that, the *sf* shall be recorded as 10_2.

4.4 AUDIO MODE Command

The AUDIO MODE control command specifies the recording mode format(s) for audio signals to be recorded by the Tape recorder/player subunit. The AUDIO MODE control command may specify formats for up to four audio blocks, as illustrated by Figure 16 below:

	msb							lsb
opcode	AUDIO MODE (71_{16})							
operand[0]	audio_ctrl[3]			audio_ctrl[2]		audio_ctrl[1]		audio_ctrl[0]
operand[1]	audio_channels[1]			sample_freq[1]		audio_channels[0]		sample_freq[0]
operand[2]	audio_channels[3]			sample_freq[3]		audio_channels[2]		sample_freq[2]

Figure 16 – AUDIO MODE control command format

NOTE – A Tape recorder/player subunit may record audio signals in three fundamental modes: normal, audio insert and AV insert. These modes are described in more detail in 4.18. An AUDIO MODE control command with operand values valid for one of these modes is not necessarily valid for the others.

An audio block is defined by whether or not it is active, the number of channels to be recorded and the sampling frequency. The HD Digital VCR Conference defines four possible audio blocks named CH1, CH2, CH3 and CH4. Each of these audio blocks is identified in the AUDIO MODE operands by an index, *n*, which may vary between zero and three, inclusive, and represents the corresponding names CH1 through CH4, respectively.

The *audio_ctrl* fields control whether or not a particular audio block is active, according to the values summarized below:

Table 13 – Audio control field value

Audio control	Description
0	Record an audio signal
1	Don't record an audio signal
2	Reserved for future specification
3	Don't care

If the value of *audio_ctrl*[n] is one (or if the value of *audio_ctrl*[n] is three and the Tape recorder/player subunit elects not to record an audio signal), the values for the corresponding *audio_channels*[n] and *sample_freq*[n] fields are ignored.

The *audio_channels* fields control, for enabled audio blocks, the number or format of the audio channels recorded as defined below:

Table 14 – Audio_channels filed value

Audio channels	Description
0	One audio channel
1	Two audio channels
2	20-bit audio mode
3	Don't care

NOTE – The use of 20-bit audio mode for one audio block forces the same value for all other enabled audio blocks. If different values are provided in the AV/C command frame, the command may be refused with a response of NOT IMPLEMENTED.

The *sample_freq* fields control, for enabled audio blocks, the sampling frequency according to the following definition:

Table 15 – Sample_freq filed value

Sampling frequency	Description
0	48 kHz
1	44.1 kHz
2	32 kHz
3	Don't care

The "don't care" value in any of the AUDIO MODE operands permits the Tape recorder/player subunit to select any permissible mode value for the operand. Default behavior of the Tape recorder/player subunit may be obtained by means of the status command form of the AUDIO MODE command, shown in Figure 17 below:

	msb							lsb
opcode	AUDIO MODE (71_{16})							
operand[0]	FF_{16}							
operand[1]								
operand[2]								

Figure 17 – AUDIO MODE status command format

The AV/C response frame returns the current status of the Tape recorder/player subunit in the operands, in the format already described in the preceding tables.

4.5 BACKWARD Command

The BACKWARD control command is used to reverse the transport mechanism a specified count of units relative to the current position. Backward motion is defined as toward the beginning of the medium. In the case of both DVCR and D-VHS VCR, the BACKWARD control command requests the Tape recorder/player subunit to be paused in playback mode immediately after reversing the transport mechanism as specified. The format of the BACKWARD control command is illustrated by Figure 18 below:

	msb							lsb
opcode	BACKWARD (56_{16})							
operand[0]	measurement_unit							
operand[1]	count							
operand[2]	FF_{16}							
operand[3]								
operand[4]								

Figure 18 – BACKWARD control command format

The *measurement_unit* field may take on a value between zero and FE_{16}, inclusive. The value of *measurement_unit* determines the countable units, as defined by the table below:

Table 16 – definition of *measurement_unit*

value	measurement_unit	DVCR	D-VHS VCR	VHS VCR	8mm VCR
0	video frame	Valid	Valid	Valid	Valid
1	video scene	Valid	Valid	Invalid	Valid
2	VISS	Invalid	Valid	Valid	Invalid
3	GOP for MPEG recording	Valid	Valid	Invalid	Invalid
4	Index	Valid	Valid	Invalid	Valid
5	Skip	Valid	Valid	Invalid	Invalid
6	Photo/Picture	Valid	Invalid	Invalid	Invalid
7	*Program_start*	Invalid	Valid	Invalid	Invalid
8	*Random_marker*	Invalid	Valid	Invalid	Invalid

– A value of zero is defined as a *video frame*. In the case of DVCR, this value indicates the start position of the video frame recorded on the medium. In the case of D-VHS VCR for tracking, this value indicates the control signal recorded on the medium. In the case of VHS VCR, this value indicates the control pulse, which is recorded on the linear control track of the medium in accordance with every video frame for (head) tracking.

– A value of one is defined as a *video scene*. This value indicates the discontinuity of *REC_DATE* or *REC_TIME* data recorded on the medium.

– A value of two is defined as *VHS Index Search System(VISS)*. VISS is the position marker used to search for the starting position of a recorded program, or for a user-specified position in the program. This value is valid in the D-VHS VCR and VHS VCR. VISS is marked on a control track by modifying the duty of control pulse for tracking in record or playback. VISS is invalid in DVCR.

– A value of three is defined as *GOP for MPEG recording*. The GOP indicates the picture data group with the intra-frame data in the MPEG recorded program. This value specifies a start position of GOP on the medium.

– A value of four is defined as *Index*. Index indicates the position marker used to search for the starting position of a recorded program, or a user-specified position in the program.

  In the case of DVCR, the Index ID defined by the HD Digital VCR Conference is valid as the index.

  In the case of D-VHS VCR, the Index flag recorded on the medium is valid as the index.

  The Index flag should be recorded for five seconds at the starting point of a program, or at each starting point of index areas in that program. In case the Marker flag is valid, the Index flag specifies the index area in the program.

– A value of five is defined as *Skip*. Skip indicates the position marker for starting to skip a discarded area. In the case of DVCR, the Skip ID defined by the HD Digital VCR Conference is valid as skip. In the case of D-VHS VCR, the Skip flag recorded on the medium is valid as skip. The Skip flag should be recorded for two seconds at the starting position of a discarded area. During the skip operation, skipping can be stopped by finding a Start flag or Index flag.

– A value of six is defined as *Photo/Picture*. Photo/Picture indicates the position marker used to search for the starting position of recorded picture data(still video, photo etc). In the case of DVCR, the PP ID defined by the HD Digital VCR Conference is valid as *Photo/Picture*.

– A value of seven is defined as *program start*. Program start indicates the position marker used to search for the starting position of a program. In the case of D-VHS VCR, the Start flag recorded on the medium is valid as program start. The Start flag is recorded at the starting position of a program for five seconds.

– A value of eight is defined as *random marker*. The random marker indicates the position marker used to search for the starting position of an interesting data area specified by the user. In the case of D-VHS VCR, the Marker flag recorded on the medium is valid as the random marker. The Marker flag is recorded for two seconds at the start position of an interesting area specified by the user.

The *count* operand specifies the number of the units to be moved. In case the target can accept this command frame with its count value set to zero, the target shall return a response code of ACCEPTED, but shall not reverse the transport mechanism.

4.6 BINARY GROUP Command

A binary group is the pack format for storing private data on the DVCR medium. For complete details, please refer to the Blue Book, part 2, section D.3.5, page 211.

This command supports various combinations of operation modes {search, preset, data} with the command types {CONTROL, STATUS, SPECIFIC INQUIRY, NOTIFY, GENERAL INQUIRY}. This command is directly related to the SMPTE/EBU time code structure; a binary group pack is written immediately following a SMPTE/EBU pack on the DVCR medium.

The BINARY GROUP command with a ctype value of CONTROL is used with operation mode preset. This version of the command has the following format:

	msb							lsb
opcode	BINARY GROUP ($5A_{16}$)							
operand[0]	70_{16}							
operand[3]	BINARY GROUP 2				BINARY GROUP 1			
operand[4]	BINARY GROUP 4				BINARY GROUP 3			
operand[5]	BINARY GROUP 6				BINARY GROUP 5			
operand[6]	BINARY GROUP 8				BINARY GROUP 7			

Figure 19 – BINARY GROUP control command format (preset value)

The BINARY GROUP operands are as defined in the Blue Book reference described above.

When this control command is executed, it will establish a preset BINARY GROUP value. This value will be written to the medium when a subsequent RECORD command is executed, beginning with the first recorded frame.

If the command is successfully completed, the subunit shall return a response frame that is identical to the command frame.

The BINARY GROUP command with a *ctype* of STATUS can be used to read the preset value of a binary group, or to directly read the binary group data from the medium at the current position. It has the following format:

	msb							lsb
opcode	BINARY GROUP ($5A_{16}$)							
operand[0]	70_{16} or 71_{16}							
operand[3]	FF_{16}							
operand[4]	FF_{16}							
operand[5]	FF_{16}							
operand[6]	FF_{16}							

Figure 20 – BINARY GROUP status command format (preset value)

The stable response frame shall have the same format as the CONTROL command. It returns either the currently defined preset binary pack values or the binary pack data from the medium.

The BINARY GROUP command can be used with a ctype of SPECIFIC INQUIRY to ask about the ability to establish a preset.

The BINARY GROUP command can also be used with a ctype of NOTIFY to monitor the preset value. This means that a controller will be notified when the preset value is changed. The format of the NOTIFY command is the same as the STATUS command frame to read preset value. The response frame is also the same as the STATUS command response frame.

The following table will help to clarify the various combinations of operation mode and ctype defined for the BINARY GROUP command:

Table 17 – Combinations of operation mode and ctype

ctype	mode	valid combination?
CONTROL	search	X
	preset	YES
	data	X
STATUS	search	X
	preset	YES
	data	YES
SPECIFIC INQUIRY	search	X
	preset	YES
	data	X
NOTIFY	search	X
	preset	YES
	data	X
GENERAL INQUIRY	N/A	YES

4.7 EDIT MODE Command

The EDIT MODE control command is used to prepare the Tape recorder/player subunit for synchronized recording or playback and then to subsequently initiate synchronized recording or playback. The format of the EDIT MODE command is illustrated by Figure 21 below:

	msb							lsb
opcode	EDIT MODE (40_{16})							
operand[0]	subfunction							

Figure 21 – EDIT MODE control command format

The value of the *subfunction* operand determines the operation performed by the target, as defined by the table below:

Table 18 – EDIT MODE subfunctions

Subfunction	Value	Support level	Action
PREROLL & STANDBY for SYNC PLAY	00_{16}	O	Search to a specified position and prepare the Tape recorder/player subunit for subsequent synchronized play operations
PREROLL & STANDBY for SYNC RECORD	01_{16}	O	Search to a specified position and prepare the Tape recorder/player subunit for subsequent synchronized record operations
AREA 2+3 INSERT SYNC RECORD	21_{16}	O	Commence synchronized recording in the specified insert mode
AREA 1 INSERT SYNC RECORD	22_{16}	O	
AREA 1+2+3 INSERT SYNC RECORD	23_{16}	O	
AREA 3 INSERT SYNC RECORD	24_{16}	O	
AREA 2 INSERT SYNC RECORD	26_{16}	O	
AREA 1+2 INSERT SYNC RECORD	27_{16}	O	
AREA 1+3 INSERT SYNC RECORD	28_{16}	O	
SYNC RECORD	25_{16}	O	Commence synchronized recording
SYNC PLAY	35_{16}	O	Commence synchronized playback
N-FRAME RECORD AREA 2+3	41_{16}	O	Initiates the N-FRAME recording action, where the number of frames that will be recorded was previously set up with the PRESET command, using *parameter_ID* 05 (NUMBER OF FRAMES). Before issuing EDIT MODE 41_{16}, the controller must set up a start point for AREA 2 and AREA 3, as described in the PRESET command description
N-FRAME RECORD ALL AREA	45_{16}	O	Performs the same operation as subfunction 41_{16}, but with all areas instead of only areas 2 and 3. The same rules apply for establishing a start point, and the number of frames to record using the PRESET command

The recordable area of tape medium may be divided into eight separate areas, from AREA 0 to AREA 7. The number of recordable areas is determined by the kind of medium, such as DVCR, D-VHS and VHS, as shown in the following table:

Table 19 – Rec_area definition

Rec_area	Description			
	DVCR	D-VHS	VHS	8mm
AREA 0	ITI sector	Control track	Control track	not defined
AREA 1	Audio sector	Linear audio track	Linear audio track	PCM audio area
AREA 2	Video sector	Main code area	Video track	Video (+AFM) area
AREA 3	Subcode sector	Subcode area	not defined	Subcode area
AREA 4	not defined	not defined	not defined	not defined
AREA 5				
AREA 6				
AREA 7				

The subfunction AREA 2+3 indicates the combination of AREA 2 and AREA 3. The valid combinations of the *rec_area* are defined by the table below.

Table 20 – Combination of *rec_area*

	DVCR	D-VHS VCR	VHS VCR	8mm VCR
AREA 2+3	Video sector + Subcode sector	Main code area + Subcode area	Not defined	Video area + Subcode area
AREA 1+2+3	Video sector + Audio sector + Subcode sector	Linear audio track + Main code area + Subcode area	Not defined	Video area + PCM audio area + Subcode area
AREA 1+2	Audio sector + Video sector	Linear audio track + Main code area	Linear audio track + Video track	PCM audio area + Video area
AREA 1+3	Audio sector + Subcode sector	Linear audio track + Subcode area	Not defined	PCM audio area + Subcode area

Before one or more EDIT MODE control commands can be issued to a target Tape recorder/player subunit, the parameters used by EDIT MODE should be established by means of the PRESET control command, as described in section 4.16.

After the appropriate parameters are established by the PRESET control command, the controller sends an EDIT MODE control command with a subfunction of PREROLL & STANDBY to the Tape recorder/player subunit. The Tape recorder/player subunit searches the medium for the time code established for the START POINT, or searches for the RTC established for the RTC START POINT, and positions the transport mechanism so that a subsequent synchronized record or playback control command may be accepted within the preroll limit.

Inside an editing VCR, there are two very different mechanical modes: edit-based and play-based. In general, switching between these two modes may take some time because the operations involved in carrying out their actions are very different from each other. To improve mechanical performance, it is beneficial to have a separate action for standby to record and standby to play.

Normally there is a significant time delay between the receipt of a PREROLL & STANDBY subfunction and the Tape recorder/player subunit's readiness to accept a synchronized record or playback control command. The Tape recorder/player subunit normally returns a response of INTERIM to the EDIT MODE control command with the

PREROLL & STANDBY subfunction and later returns a response of ACCEPTED when the standby condition has been achieved.

Once in a standby state, the Tape recorder/player subunit is ready to accept an EDIT MODE control command that specifies one of the synchronized record or playback subfunctions. This causes the Tape recorder/player subunit to commence recording or playback at the START POINT or RTC START POINT after the PREROLL TIME period has elapsed. Recording or playback continues until the STOP POINT previously established.

For the N-FRAME RECORD commands, after a recording operation has been stopped, the next start point shall automatically be preset by the subunit by adding the current “start time” preset value to the “number of frames” preset value.

The preceding describes the operation of the EDIT MODE command with a *ctype* of CONTROL. The same *opcode* may also be used with a *ctype* of STATUS, in which case a status response frame is returned. The format of the EDIT MODE status command and its associated AV/C response frame are illustrated by Figure 22 and Figure 23 below:

	msb							lsb
opcode	EDIT MODE (40_{16})							
operand[0]	FF_{16}							

Figure 22 – EDIT MODE status command format

	msb							lsb
opcode	EDIT MODE (40_{16})							
operand[0]	edit_mode							

Figure 23 – EDIT MODE status response format

The values returned for *edit_mode* are the same as defined for the EDIT MODE control command subfunctions in Table 18. If the Tape recorder/player subunit is not in one of the edit modes described in Table 18, a value of 60_{16} is returned for *operand[0]*. A Tape recorder/player subunit typically returns a response code of IN TRANSITION while positioning to the preroll point or during the preroll time period, and a response code of STABLE at other times while an EDIT MODE operation is in progress.

4.8 FORWARD Command

The FORWARD control command is used to advance the transport mechanism a specified count of units relative to the current position. Forward motion is defined as away from the beginning of medium. In the case of both DVCR and D-VHS VCR, the FORWARD control command requests the Tape recorder/player subunit to be paused in playback mode immediately after advancing the transport mechanism the specified count of units relative to the current position. The format of the FORWARD control command is illustrated by Figure 24 below.

	msb							lsb
opcode	FORWARD (55_{16})							
operand[0]	measurement_unit							
operand[1]	count							
operand[2]	FF_{16}							
operand[3]								
operand[4]								

Figure 24 – FORWARD control command format

The *measurement_unit* field may take on a value between zero and FE_{16}, inclusive. The value of the *measurement_unit* determines the countable units, as defined in the BACKWARD command described above.

The *count* field specifies the number of the units to be advanced. In case the target can accept this command frame with its *count* value set to zero, the target shall return a response code of ACCEPTED, but shall not advance the transport mechanism.

4.9 INPUT SIGNAL MODE Command

The INPUT SIGNAL MODE control command configures a Tape recorder/player subunit to accept its input data in one of the formats defined by the DVCR, D-VHS, S-VHS, VHS, 8mm, MICROMV, Analog Audio Cassette, and HDV Specifications. The structure of the INPUT SIGNAL MODE command is shown in Figure 25 below:

	msb							lsb
opcode	INPUT SIGNAL MODE (79_{16})							
operand[0]	signal_mode							

Figure 25 – INPUT SIGNAL MODE control command format

The *signal_mode* field specifies the characteristics of the input data, as defined by the table that follows:

Table 21 – Signal modes

Value	Signal mode	Reference
00_{16}	SD 525-60	DVCR Specification [8] / HDV Specification [15]
04_{16}	SDL 525-60	
08_{16}	HD 1125-60	
80_{16}	SD 625-50	
84_{16}	SDL 625-50	
88_{16}	HD 1250-50	
10_{16}	MPEG 25Mbps-60 (DVCR) / HD1 mode-60 (HDV)	
14_{16}	MPEG 12.5Mbps-60	
18_{16}	MPEG 6.25Mbps-60	
90_{16}	MPEG 25Mbps-50 (DVCR) / HD1 mode-50 (HDV)	
94_{16}	MPEG 12.5Mbps-50	
98_{16}	MPEG 6.25Mbps-50	

Value	Signal mode	Reference
01_{16}	D-VHS Digital	D-VHS STANDARD [12]
05_{16}	Analog VHS NTSC 525/60	VHS STANDARD [13]
25_{16}	Analog VHS M-PAL 525/60	
$A5_{16}$	Analog VHS PAL 625/50	
$B5_{16}$	Analog VHS N-PAL 625/50	
$C5_{16}$	Analog VHS SECAM 625/50	
$D5_{16}$	Analog VHS ME-SECAM 625/50	
$0D_{16}$	Analog S-VHS 525/60	S-VHS STANDARD [14]
ED_{16}	Analog S-VHS 625/50	
06_{16}	Analog 8mm NTSC	8mm Specification [9]
86_{16}	Analog 8mm PAL	
$0E_{16}$	Analog Hi8 NTSC	
$8E_{16}$	Analog Hi8 PAL	
24_{16}	MPEG 12.5Mbps-60	MICROMV
28_{16}	MPEG 6.25Mbps-60	
$A4_{16}$	MPEG 12.5Mbps-50	
$A8_{16}$	MPEG 6.25Mbps-50	
20_{16}	Audio	Analog Audio Cassette
$1A_{16}$	HD2 mode-60	HDV Specification [15]
$9A_{16}$	HD2 mode-50	

The INPUT SIGNAL MODE command may also be used to query which signal format the Tape recorder/player subunit is currently configured to accept. In this case, the *ctype* field shall be STATUS and the command format illustrated by Figure 26 below is used:

	msb							lsb
opcode	INPUT SIGNAL MODE (79_{16})							
operand[0]	FF_{16}							

Figure 26 – INPUT SIGNAL MODE status command format

The AV/C response frame returned by the Tape recorder/player subunit updates *operand[0]* with one of the values described in Table 21 above.

The INPUT SIGNAL MODE command may also be used as a notify command. The notify command has the same syntax as the status command, but with a ctype of NOTIFY. A notification shall be returned by the target to the controller that issued the notify command when there is a change in the configuration of input signal mode of the VCR subunit. The notify response has the same format as the status response frame.

4.10 LOAD MEDIUM Command

The LOAD MEDIUM control command is used to control the loaded or unloaded state of medium in the transport mechanism. The format of the LOAD MEDIUM command is illustrated by Figure 27 below:

	msb							lsb
opcode	LOAD MEDIUM ($C1_{16}$)							
operand[0]	subfunction							

Figure 27 – LOAD MEDIUM control command format

The value of the *subfunction* operand determines the operation performed by the target, as defined by the table below.

Table 22 – LAOD MEDIUM subfunctions

subfunction	Value	Support level	Action
EJECT	60_{16}	O	Eject the cassette from the Tape recorder/player subunit
OPEN TRAY	31_{16}	O	Open the tray
CLOSE TRAY	32_{16}	O	Close the tray

4.11 MARKER command

The MARKER control command is used to record or erase a marker signal on the medium. The format of the MARKER control command is illustrated by the figure below:

	msb							lsb
opcode	MARKER (CA_{16})							
operand[0]	mk_er	marker_type						

Figure 28 – MARKER control command format

The *mk_er* field controls whether or not a specified position is active. A value of zero is to erase a marker signal while a value of one is to record a marker signal. The *marker_type* indicates the kind of marker signal to be marked or erased, as defined by the table below:

Table 23 – Marker types

Value	marker_type
0000001_2	VISS

Recording or erasing a marker signal should be immediately executed as soon as the control command is accepted by the Tape recorder/player subunit. Usually, in the case of marking, this control command should be issued to the Tape recorder/player subunit after defining the specified position; the mark will be made at the current medium position. In the case of erasing, this control command should be issued to the Tape recorder/player subunit after finding the specified position on the medium. The Tape recorder/player subunit, which accepted the control command, shall not return a response of INTERIM to prevent the specified position from shifting. If an erase marker control command is issued and there is no marker at the current location, then the Tape recorder/player subunit shall ACCEPT the command.

The MARKER command may also be used with a *ctype* value of STATUS. The MARKER status command is used in playback to query whether or not the recorded marker, which is specified by the *marker_type*, exists at this point on the medium. The format of the MARKER status command is illustrated below:

	msb							lsb
opcode	MARKER (CA_{16})							
operand[0]	FF_{16}							

Figure 29 – MARKER status command format

The status and notify response frame is the same as the control command frame. In the response frame returned after a MARKER status or notify command, the *mk_er* value indicates whether or not the recorded marker signal specified by the *marker_type* exists at this point on the medium. A value of zero indicates that the marker signal does not exist while a value of one indicates that the marker signal does exist.

The MARKER command may also be used as a notify command. The command frame for NOTIFY is the same as the command frame for STATUS. The MARKER notification service is helpful for controllers who are monitoring a subunit to find the positions where the marker signals are recorded. For example, in order to make a list of the marked positions, along with an accurate indication of these positions, the controller should do the following:

1) Make sure that the medium is at a position, which does NOT have a marker.

2) Issue the MARKER notify command. When the next marked position is found, the Tape recorder/player subunit shall return a response of CHANGED.

3) After receiving the response of CHANGED, the controller can get the accurate position by issuing the ABSOLUTE TRACK NUMBER status command, the TIME CODE status command, or the RELATIVE TIME COUNTER status command.

4.12 MEDIUM INFO Command

The MEDIUM INFO status command is used to request information about the cassette currently inserted in a Tape recorder/player subunit. The format of the MEDIUM INFO status command is illustrated by Figure 30 below:

	msb							lsb
opcode	MEDIUM INFO (DA_{16})							
operand[0]	$7F_{16}$							
operand[1]	$7F_{16}$							

Figure 30 – MEDIUM INFO status command format

The information returned is formatted in a response frame, as shown by Figure 31 below:

	msb							lsb
opcode	MEDIUM INFO (DA_{16})							
operand[0]	cassette_type							
operand[1]	tape_grade_and_write_protect							

Figure 31 – MEDIUM INFO response format

The first operand returned, *cassette_type*, encodes information about the kind of cassette present in the Tape recorder/player subunit. If the cassette type is recognized, *tape_grade_and_write_protect* encodes additional information about the cassette. Valid combinations of *cassette_type* and *tape_grade_and_write_protect* are defined below:

NOTE – Both cassette_types of VHS compact (VHS-C) and S-VHS compact (S-VHS-C) cassettes in a VHS or a S-VHS cassette adapter become VHS cassette(22_{16}) when those cassettes in the adapter are inserted into the D-VHS VCR.

Table 24 – Cassette types

Cassette type	Write protect	Comment
DVCR standard cassette (31_{16})	30_{16}	OK to record on medium
	31_{16}	Recording inhibited
	40_{16}	MP tape and OK to record on medium
	41_{16}	MP tape and recording inhibited
DVCR small cassette (32_{16})	30_{16}	OK to record on medium
	31_{16}	Recording inhibited
	40_{16}	MP tape and OK to record on medium
	41_{16}	MP tape and recording inhibited
DVCR medium cassette (33_{16})	40_{16}	MP tape and OK to record on medium
	41_{16}	MP tape and recording inhibited
VHS cassette (22_{16})	30_{16}	VHS tape and OK to record on medium
	31_{16}	VHS tape and recording inhibited
	40_{16}	S-VHS tape and OK to record on medium
	41_{16}	S-VHS tape and recording inhibited
	50_{16}	D-VHS tape and OK to record on medium
	51_{16}	D-VHS tape and recording inhibited
VHS-C cassette (23_{16})	30_{16}	VHS tape and OK to record on medium
	31_{16}	VHS tape and recording inhibited
	40_{16}	S-VHS tape and OK to record on medium
	41_{16}	S-VHS tape and recording inhibited
No cassette present (60_{16})	$7F_{16}$	
Unknown cassette ($7E_{16}$)	$7F_{16}$	Write protect status cannot be determined
8mm cassette (12_{16})	30_{16}	8mm MP tape and OK to record on medium
	31_{16}	8mm MP tape and recording inhibited
	40_{16}	8mm ME tape and OK to record on medium
	41_{16}	8mm ME tape and recording inhibited
	50_{16}	Hi8 MP tape and OK to record on medium
	51_{16}	Hi8 MP tape and recording inhibited
	60_{16}	Hi8 ME tape and OK to record on medium
	61_{16}	Hi8 ME tape and recording inhibited
MICROMV cassette (41_{16})	30_{16}	OK to record on medium
	31_{16}	Recording inhibited
Analog audio cassette (01_{16})	30_{16}	OK to record on medium
	31_{16}	Recording inhibited

4.13 OPEN MIC Command

The OPEN MIC control command is used to gain access to the nonvolatile memory that may be part of a DVCR cassette. The format of the OPEN MIC control command is illustrated by Figure 32 below:

	msb							lsb
opcode	OPEN MIC (60_{16})							
operand[0]	subfunction							

Figure 32 – OPEN MIC control command format

The value of the *subfunction* operand determines the operation performed by the target, as defined by the table below.

Table 25 – OPEN MIC subfunctions

Subfunction	Value	Support level	Action
CLOSE	00_{16}	R	Relinquish use of the MIC
READ OPEN	01_{16}	R	Open the MIC for read-only access
WRITE OPEN	03_{16}	O	Open the MIC for read or write access

A Tape recorder/player subunit shall respond to OPEN MIC control commands as follows:

– After a power reset, command reset or Serial Bus reset, the MIC of any inserted cassette shall be in a closed state. If a cassette is inserted and is not write-protected, the Tape recorder/player subunit shall accept an OPEN MIC control command from any controller.

– The Tape recorder/player subunit shall reject an OPEN MIC control command if no cassette is inserted, if an inserted cassette has no MIC or if an inserted cassette is write-protected and the OPEN MIC command *subfunction* specifies WRITE OPEN.

– If a cassette MIC is closed or has only been opened for read access, a Tape recorder/player subunit may accept any number of OPEN MIC requests with a *subfunction* of READ OPEN.

– If a cassette MIC is closed or has only been opened for read access, a VCR unit may accept a single OPEN MIC control command with a *subfunction* of WRITE OPEN. This OPEN MIC operation for write access forces any existing read only opens to be closed.

– If a cassette MIC is open for write access, a Tape recorder/player subunit shall reject any OPEN MIC control commands except those with a *subfunction* of CLOSE sent by the controller that opened the cassette MIC for write access.

– A Tape recorder/player subunit shall implement a time-out period, recommended to be longer than one minute since the last accepted OPEN MIC, READ MIC or WRITE MIC control command. If this time-out period expires when a cassette MIC is open for write access, the Tape recorder/player subunit shall accept an OPEN MIC control command from any controller. This forces the existing write open to be closed.

After a Serial Bus reset, it is desirable for controllers that previously have opened MIC to reestablish their opened status. This is accomplished by cooperation amongst controllers. Any controller that has previously opened a MIC may issue an OPEN MIC control command immediately after a bus reset. Controllers that have not opened a MIC prior to the bus reset are expected to delay at least two seconds before issuing any OPEN MIC control commands.

NOTE – Because other control commands or manual operations may have side-effects that affect access to cassette MIC, Tape recorder/player subunits are expected to reject such requests while a cassette MIC is open for either read or write access.

The OPEN MIC command may also be used with a *ctype* value of STATUS to determine the current closed or open status of a MIC. The format of this status command is illustrated by Figure 33 below:

	msb							lsb
opcode	OPEN MIC (60_{16})							
operand[0]	FF_{16}							

Figure 33 – OPEN MIC status command format

The response frame returned by the Tape recorder/player subunit updates *operand[0]* to reflect the current state of the MIC or cassette, as summarized below:

Table 26 – OPEN MIC states

Value	Meaning
00_{16}	MIC closed
01_{16}	MIC opened for read-only access by one or more controllers and able to accept additional read-only OPEN MIC commands
04_{16}	No MIC cassette inserted
08_{16}	No cassette inserted
11_{16}	MIC opened for read-only accesses and unable to accept additional read-only OPEN MIC commands
33_{16}	MIC opened for read and write accesses

4.14 OUTPUT SIGNAL MODE Command

The OUTPUT SIGNAL MODE control command configures a Tape recorder/player subunit to transmit its output data in one of the formats defined by the DVCR, D-VHS, S-VHS, VHS, 8mm, MICROMV, Analog Audio Cassette, and HDV Specifications. The structure of the OUTPUT SIGNAL MODE command is shown in Figure 34 below:

	msb							lsb
opcode	OUTPUT SIGNAL MODE (78_{16})							
operand[0]	signal_mode							

Figure 34 – OUTPUT SIGNAL MODE control command format

The *signal_mode* field specifies the characteristics of the output data, as defined by Table 21.

The OUTPUT SIGNAL MODE command may also be used to query which signal format the Tape recorder/player subunit is currently configured to transmit. In this case, the *ctype* field shall be STATUS and the command format illustrated by Figure 35 below is used.

	msb							lsb
opcode	OUTPUT SIGNAL MODE (78_{16})							
operand[0]	FF_{16}							

Figure 35 – OUTPUT SIGNAL MODE status command format

The AV/C response frame returned by the Tape recorder/player subunit updates *operand[0]* with one of the values described in Table 21 above.

The OUTPUT SIGNAL MODE command may also be used as a notify command. The notify command has the same syntax as the status command, but with a ctype of NOTIFY. A notification shall be returned by the target to the controller that issues the notify command when there is a change in the configuration of output signal mode of the VCR subunit. The notify response has the same format as the status response frame.

4.15 PLAY Command

The PLAY control command is used to request the VCR transport mechanism to playback data previously recorded on the medium. The format of the PLAY control command is illustrated by Figure 36 below.

	msb							lsb
opcode	PLAY ($C3_{16}$)							
operand[0]	playback_mode							

Figure 36 – PLAY control command format

The value of *operand[0]* determines the playback mode used by the target, as defined by Table 27.

The subunit support level for PLAY, mandatory (M), recommended (R), or optional (O), varies according to the playback mode requested.

If no cassette is loaded, the Tape recorder/player subunit shall refuse the PLAY control command and return a *response* of REJECTED.

Speed variations in either a forward or reverse playback direction are collectively referred to as trick play modes. Except for FASTEST FORWARD and FASTEST REVERSE, a Tape recorder/player subunit is not required to support any of the trick play modes. However, if trick play modes are supported a Tape recorder/player subunit shall implement them as follows:

There are four groups of trick play modes: slow forward, fast forward, slow reverse and fast reverse. A Tape recorder/player subunit may implement each group independently.

a) If a Tape recorder/player subunit implements a trick play group, it shall implement the basic playback option, *i.e.*, either SLOWEST or FASTEST in the direction implemented. PLAY control commands with an *operand[0]* that specifies a SLOW *n* or FAST *n* playback mode may be rejected by the Tape recorder/player subunit as NOT IMPLEMENTED. Optionally, the Tape recorder/player subunit may accept all of the SLOW *n* or FAST *n* operands and interpret them as SLOWEST or FASTEST.

b) If a Tape recorder/player subunit implements more than one speed within a trick play group, it shall recognize all of the SLOW *n* or FAST *n* playback modes as well as the SLOWEST or FASTEST playback mode. A Tape recorder/player subunit is not required to implement all seven possible playback speeds; it is required only to map all possible playback modes within the trick play group to the speeds it does support. The actual speeds encoded by the playback modes shall be subject to one of the following restrictions, as appropriate:

SLOWEST <= SLOW 6 <= SLOW 5 <= SLOW 4 <= SLOW 3 <= SLOW 2 <= SLOW 1 <= X1

or

X1 <= FAST 1 <= FAST 2 <= FAST 3 <= FAST 4 <= FAST 5<= FAST 6 <= FASTEST

Table 27 – Playback modes

Playback Mode	Value	Support level	Description
NEXT FRAME	30_{16}	R	Playback the next sequential frame or field
SLOWEST FORWARD	31_{16}	R	Playback at a special effect speed described in detail below
SLOW FORWARD 6	32_{16}	O	
SLOW FORWARD 5	33_{16}	O	
SLOW FORWARD 4	34_{16}	O	
SLOW FORWARD 3	35_{16}	O	
SLOW FORWARD 2	36_{16}	O	
SLOW FORWARD 1	37_{16}	O	
X1	38_{16}	O	Playback at normal speed
FAST FORWARD 1	39_{16}	O	Playback at a special effect speed described in detail below
FAST FORWARD 2	$3A_{16}$	O	
FAST FORWARD 3	$3B_{16}$	O	
FAST FORWARD 4	$3C_{16}$	O	
FAST FORWARD 5	$3D_{16}$	O	
FAST FORWARD 6	$3E_{16}$	O	
FASTEST FORWARD	$3F_{16}$	M	
PREVIOUS FRAME	40_{16}	R	Playback the previous sequential frame or field
SLOWEST REVERSE	41_{16}	R	Playback in reverse at a special effect speed described in detail below
SLOW REVERSE 6	42_{16}	O	
SLOW REVERSE 5	43_{16}	O	
SLOW REVERSE 4	44_{16}	O	
SLOW REVERSE 3	45_{16}	O	
SLOW REVERSE 2	46_{16}	O	
SLOW REVERSE 1	47_{16}	O	
X1 REVERSE	48_{16}	O	Playback at normal speed in reverse
FAST REVERSE 1	49_{16}	O	Playback in reverse at a special effect speed described in detail below
FAST REVERSE 2	$4A_{16}$	O	
FAST REVERSE 3	$4B_{16}$	O	
FAST REVERSE 4	$4C_{16}$	O	
FAST REVERSE 5	$4D_{16}$	O	
FAST REVERSE 6	$4E_{16}$	O	
FASTEST REVERSE	$4F_{16}$	M	
REVERSE	65_{16}	O	Playback at normal speed in reverse
REVERSE PAUSE	$6D_{16}$	O	Pause in reverse playback
FORWARD	75_{16}	M	Playback at normal speed
FORWARD PAUSE	$7D_{16}$	M	Pause in playback

4.16 PRESET Command

The PRESET control command is used to set or reset values of internal parameters maintained by a Tape recorder/player subunit. The format of the PRESET command is illustrated by Figure 37 below.

	msb							lsb
opcode	PRESET (45_{16})							
operand[0]	reset	parameter_ID						
operand[1]	parameter_value							
operand[2]								
operand[3]								
operand[4]								

Figure 37 – PRESET control command format

The PRESET command is used in conjunction with the EDIT MODE command to prepare for and execute editing operations. For more details, please refer to the EDIT MODE command description.

If the controller wishes to establish a certain value for a preset, then it shall clear the *reset* bit and pass in the appropriate values for the specified *parameter_ID*.

The value of *parameter_ID* determines which preset is initialized and how the four ***parameter_value*** bytes are interpreted, as described in the following table:

Table 28 – Parameter IDs

Parameter ID / Description	Parameter value				
	Value	operand[1]	operand[2]	operand[3]	operand[4]
START POINT	00_{16}	frame	second	minute	hour
STOP POINT	01_{16}				
PREROLL TIME	02_{16}			FF_{16}	FF_{16}
RTC START POINT	03_{16}	sign and frame		minute	hour
RTC STOP POINT	04_{16}				
NUMBER OF FRAMES	05_{16}	(msb)	binary value		(lsb)
AREA 1 START POINT	10_{16}	frame	second	minute	hour
AREA 1 STOP POINT	11_{16}				
AREA 2 START POINT	12_{16}				
AREA 2 STOP POINT	13_{16}				
AREA 3 START POINT	14_{16}				
AREA 3 STOP POINT	15_{16}				
ALL	$7F_{16}$	FF_{16}	FF_{16}	FF_{16}	FF_{16}

The *parameter_ID* operand can take one of the following values:

– START POINT (00_{16}) is defined to mean “All AREAS” or the default area.

– STOP POINT (01_{16}) is defined to mean “All AREAS” or the default area.

– In some cases, a subunit may not have the ability to record each of the areas. For example, it may be able to record AREA 2, but not AREA 1 or AREA 3. However, it can play back all areas.

- This is why *parameter_ID* 00_{16} and 01_{16} are defined as “default area” operands. When the controller issues these *parameter_ID* values, then all *recordable* areas will be preset for start or stop times, even though it may not be ALL areas that are preset.

- PREROLL TIME (02_{16}) is the time that a Tape recorder/player subunit prerolls.

- NUMBER OF FRAMES (05_{16}) means “record this number of frames from the preset start time”. The MSB of the frame count is in operand[1], and the LSB is in operand[4].

- AREA 1 START POINT (10_{16}) establishes the start point only for AREA 1 recording.

- AREA 1 STOP POINT (11_{16}) establishes the stop point only for AREA 1 recording.

- AREA 2 START POINT (12_{16}) establishes the start point only for AREA 2 recording.

- AREA 2 STOP POINT (13_{16}) establishes the stop point only for AREA 2 recording.

- AREA 3 START POINT (14_{16}) establishes the start point only for AREA 3 recording.

- AREA 3 STOP POINT (15_{16}) establishes the stop point only for AREA 3 recording.

- The AREA start and stop points can be individually set. For some EDIT MODE commands in which multiple areas are affected simultaneously (e.g. AREA 2 + 3 INSERT SYNC RECORD), it will be necessary to issue several PRESET commands, each specifying an individual area start and stop point.

- When the AREA start and stop points are set, they must be set consistently; all start points must be the same, and all stop points must be the same. The stop point must have a value that is greater than the start point. The resulting action when these rules are not followed is up to the subunit implementation; for example, if the stop points for two areas do not match, then maybe neither stop point is honored, or maybe the highest value stop point is honored. It is up to the manufacturer to decide what to do in these situations.

- ALL PARAMETERS ($7F_{16}$) means to affect all of the parameters with the command. Parameters are the ALL AREAS start and stop points, the individual area start and stop points, the preroll and the number of frames. The value $7F_{16}$ is only defined for the command when the *reset* field is set to 1; it does not make sense to define it for the command when the *reset* field is set to 0.

The time values *frame*, *second*, *minute* and *hour* are all encoded in binary coded decimal (BCD) format where each nibble of a byte represents a decimal digit. The *minute* and *hour* values are not used for PREROLL TIME. The most significant bit of *sign and frame* shows minus or plus sign. A value of zero indicates plus sign while a value of one indicates minus sign. The value of the least significant seven bits of *sign and frame* field is encoded in binary coded decimal (BCD) format where upper three bits and lower four bits represent decimal digits.

If a Tape recorder/player subunit implements the PRESET control command but does not support a particular *parameter_value* for one of the parameters, it may generate a REJECTED response to the command, or it may ACCEPT the command and ignore the unsupported *parameter_value*.

NOTE – One example of this is given by Tape recorder/player subunits that implement fixed preroll times. If a PRESET control command to set the preroll time specifies frame and second values different from the fixed preroll time, the Tape recorder/player subunit shall reject the command.

When the ***reset*** field is set in the control command, the preset specified by *parameter_ID* shall be set to its default value if such a value exists. When the controller wishes to cause a reset, it shall pass in values of FF_{16} for the *parameter_value* operand. If the specified *parameter_ID* does not have a default value, then the target shall “clear” the value of the specified *parameter_ID*. The meaning of “clear” is implementation dependent, but it means that the target shall remove the current value of that *parameter_ID*.

The PRESET command may also be used with a *ctype* value of STATUS. This status command, whose format is shown by Figure 38 below, reports either the current value or the default value of the specified parameter in the response frame returned. The response frame has the same format as the control command frame.

	msb							lsb
opcode	PRESET (45_{16})							
operand[0]	reset	parameter_ID						
operand[1]	FF_{16}							
…								
operand[4]								

Figure 38 – PRESET status command format

When the *reset* field is set to 0 in the status command frame, then the controller is asking for the current value of the specified *parameter_ID*. The current value of the parameter is reported by the *operand[n]* bytes returned in the response frame. If the current value is NOT the default value, then the *reset* field in the response frame shall be 0. If the current value IS the default value, then the reset operand in the response frame shall be 1. If the *parameter_ID* does not have a default value, but it has been reset, then the response frame shall contain FF FF FF FF_{16} for the *parameter_value* operand; otherwise, it shall contain the actual default value.

When the *reset* field is set to 1 in the status command frame, then the controller is asking for the default value of the specified *parameter_ID*. In the response frame, the *reset* field shall be 1, and the *parameter_value* operand shall either be FF FF FF FF_{16} in the case where no default exists for the specified *parameter_ID*, or it shall be the value of the default for the *parameter_ID* if a default is defined.

4.17 READ MIC Command

The READ MIC control command is used to access variable-length data stored in cassette nonvolatile memory (MIC). A controller is expected to have issued an OPEN MIC control command to the target subunit before performing any MIC read or write operations. The format of the READ MIC control command is illustrated by Figure 39 below:

	msb							lsb
opcode	READ MIC (61_{16})							
operand[0]	data_length							
operand[1]	(most significant byte)					MIC_address		
operand[2]	(least significant byte)							

Figure 39 – READ MIC control command format

The *data_length* field specifies the number of bytes to be read from the MIC.

The *MIC_address* field specifies the starting offset of the data within the MIC.

– If an ACCEPTED response frame is returned by the target after a READ MIC control command, the response data consists of additional operands, *operand[3]* through *operand[data_length + 2]*, that contain the data bytes requested.

– If the Tape recorder/player subunit is not be able to return the number of bytes indicated by *data_length* in a single operation, it shall return the maximum quantity of data it is able and adjust the *data_length* field in the response frame accordingly.

– If the combination of *MIC_address* and *data_length* address unimplemented memory, the Tape recorder/player subunit may return the addressable data and adjust the *data_length* field in the response frame accordingly.

4.18 RECORD Command

The RECORD control command is used to request the VCR transport mechanism to record signal(s) on the medium. The format of the RECORD control command is illustrated by Figure 40 below:

	msb							lsb
opcode	RECORD ($C2_{16}$)							
operand[0]	recording_mode							

Figure 40 – RECORD control command format

The value of *operand[0]* determines the recording mode used by the target, as defined by the table below:

Table 29 – Recording modes

Recording Mode	Value	Support level	Description
AREA 2+3 INSERT	31_{16}	R	Replace the specified type of signal with a new signal but leave the other signals(s) on the medium intact
AREA 1 INSERT	32_{16}	R	
AREA 1+2+3 INSERT	33_{16}	R	
AREA 3 INSERT	34_{16}	O	
AREA 2 INSERT	36_{16}	O	
AREA 1+2 INSERT	37_{16}	O	
AREA 1+3 INSERT	38_{16}	O	
AREA 2+3 INSERT PAUSE	41_{16}	R	Pause recording signal(s) on the medium and establish the recording mode indicated
AREA 1 INSERT PAUSE	42_{16}	R	
AREA 1+2+3 INSERT PAUSE	43_{16}	R	
AREA 3 INSERT PAUSE	44_{16}	O	
AREA 2 INSERT PAUSE	46_{16}	O	
AREA 1+2 INSERT PAUSE	47_{16}	O	
AREA 1+3 INSERT PAUSE	48_{16}	O	
RECORD	75_{16}	M	Overwrite all signal(s) on the medium
RECORD PAUSE	$7D_{16}$	M	Pause while recording all signal(s)

Each insert recording in the *rec_area* such as AREA 1, AREA 2, and AREA 3, or in the combination of these areas is performed when the *recording_mode* field takes on a value between 31_{16} and 48_{16} inclusive, except 35_{16}, 39_{16} - $3F_{16}$, 40_{16} and 45_{16}.

The definitions of *rec_area* combinations for each type of Tape recorder/player subunit are presented in the EDIT MODE command description.

The subunit support level for RECORD, mandatory (M), recommended (R), or optional (O), varies according to the recording mode requested.

NOTE – If a Tape recorder/player subunit does not have record capabilities, none of the recording modes for the RECORD control command are mandatory.

If no cassette is loaded or a write-protected cassette is loaded, the Tape recorder/ player subunit shall refuse the RECORD control command and return a *response* of REJECTED.

4.19 RECORDING DATE Command

The RECORDING DATE command with a *ctype* of value CONTROL is used to preset values of the recording date. The frame format is as follows:

	msb							lsb
opcode	RECORDING DATE (53_{16})							
operand[0]	70_{16}							
operand[1]	ds	tm	time_zone					
operand[2]	3		day					
operand[3]	weekday		month					
operand[4]	year							

Figure 41 – RECORDING DATE control command format

The *ds* bit indicates whether or not daylight saving time was in force at the time of recording. A value of zero indicates daylight saving time while a value of one indicates standard time. In the case of D-VHS VCR, *ds* has no information and shall have a value of one.

The *tm* bit indicates whether or not the time zone is offset by 30 minutes from Greenwich Meridian Time (GMT). A value of zero indicates that local time equals GMT plus *time_zone* plus 30 minutes while a value of one indicates that no offset is necessary.

The *time_zone* field specifies the hours offset from GMT. The *time_zone* field is encoded in binary coded decimal (BCD) format where the most significant two bits and the least significant nibble each represent a decimal digit. The *time_zone* field shall have a value between zero and 23, inclusive, or a value of $3F_{16}$ (which indicates that no time zone information is available). In the case of D-VHS VCR, if the Validity Flag specified by the REC. DATA PACK has a value of zero, the value of the *time_zone* field is $3F_{16}$ (no information).

The *day* field specifies the day of the month. The *day* field is encoded in binary coded decimal (BCD) format where the most significant two bits and the least significant nibble each represent a decimal digit. The *day* field shall have a value between one and 31, inclusive, or a value of $3F_{16}$ (which indicates that no day of the month information is available).

The *weekday* field specifies the day of the week and is encoded according to the following table:

Table 30 – Weekday field values

Value	Day of the week
0	Sunday
1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday
7	No information available

The *month* field is encoded in binary coded decimal (BCD) format where the most significant bit and the least significant nibble each represent a decimal digit. The *month* field shall have a value between one and twelve, inclusive, that represents the months January through December, respectively, or a value of $1F_{16}$ (which indicates that no month information is available).

The *year* field specifies the least significant two digits of the year, *Anno Domini* (AD). The *year* field is encoded in binary coded decimal (BCD) format where the each nibble represents a decimal digit. The *year* field shall have a value between zero and 99, inclusive, or a value of FF_{16} (which indicates that no year information is available).

The RECORDING DATE command with a ctype of STATUS has two functions. One is to read the recording date preset value, and the other is to request the Tape recorder/player subunit to return the recording date from the medium of the inserted cassette.

The STATUS command frame used to read the preset is defined as follows:

	msb							lsb
opcode	RECORDING DATE (53_{16})							
operand[0]	70_{16}							
operand[1]	FF_{16}							
operand[2]	FF_{16}							
operand[3]	FF_{16}							
operand[4]	FF_{16}							

Figure 42 – RECORDING DATE status command format (read preset value)

The stable response frame will be the same as the CONTROL command frame.

The other RECORDING DATE status command, which requests the Tape recorder/player subunit to return the recording date at the current position of the inserted cassette, has the format as illustrated by Figure 43 below.

	msb							lsb
opcode	RECORDING DATE (53_{16})							
operand[0]	71_{16}							
operand[1]	FF_{16}							
…								
operand[4]								

Figure 43 – RECORDING DATE status command format (read current value)

If the Tape recorder/player subunit is able to return a STABLE response to the RECORDING DATE status command, the AV/C response frame has the format illustrated by Figure 44 below:

	msb							lsb
opcode	RECORDING DATE (53_{16})							
operand[0]	71_{16}							
operand[1]	ds	tm	*time_zone*					
operand[2]	3		day					
operand[3]	weekday		month					
operand[4]	year							

Figure 44 – RECORDING DATE response format (read current value)

All of the operands for the above response frame are exactly as defined for the CONTROL command frame.

In addition, the RECORDING DATE command with a *ctype* of NOTIFY has RECOMMENDED support, so that controllers can be notified when the recording date of the inserted medium changes. If the recording date of the medium has been changed, and there is a pending notification request, then the target subunit shall send the notification message when the RECORD operation begins.

The format of the NOTIFY command is exactly the same as for the status response or control frames; the fields will be filled in with the newly defined information for the recording date.

4.20 RECORDING SPEED Command

The RECORDING SPEED command is used to set or query the recording speed for the Tape recorder/player subunit's transport mechanism. The format of the RECORDING SPEED control command is illustrated by Figure 45 below:

	msb							lsb
opcode	RECORDING SPEED (DB_{16})							
operand[0]	recording_speed							

Figure 45 – RECORDING SPEED control command format

The value of *operand[0]* determines the recording speed to be used by the Tape recorder/player subunit. The *recording_speed* operand shall have a value between zero and FE_{16}, inclusive. A *recording_speed* value of $6F_{16}$ represents a standard recording speed; values less than $6F_{16}$ are considered lower recording speeds while values greater than $6F_{16}$ are considered higher speeds. In both cases of DVCR and D-VHS VCR, the recording speed defines the track pitch.

Four values for *recording_speed* are defined for DVCR in the table below:

Table 31 – Recording_speed definition for DVCR

Recording Speed	Value	Comment
Speed 32	20_{16}	6.67 µm track pitch
Standard speed	$6F_{16}$	10 µm track pitch
Speed 176	$B0_{16}$	15 µm track pitch
Speed 192	$C0_{16}$	18 µm track pitch

One value for *recording_speed* is specified by the D-VHS specification, as shown in the table below:

Table 32 – Recording_speed definition for DVHS

Recording Speed	Value	Comment
Standard speed	$6F_{16}$	29 µm track pitch

Three values for *recording_speed* are specified by the VHS specification, as shown in the table below:

Table 33 – Recording_speed definition for VHS

Recording Speed	Value	Comment
Speed 33	21_{16}	EP mode
Speed 32	20_{16}	LP mode
Standard speed	$6F_{16}$	SP mode

Two values for *recording_speed* are specified by the 8mm VIDEO specification, as shown in the table below:

Table 34 – Recording_speed definition for 8mm VIDEO

Recording Speed	Value	Comment
Speed 32	20_{16}	LP mode
Standard speed	$6F_{16}$	SP mode

Table 35 – Recording_speed definition for MICROMV

Recording Speed	Value	Comment
Speed 32	20_{16}	LP mode
Standard speed	$6F_{16}$	SP mode

In addition to a *ctype* value of CONTROL, the RECORDING SPEED command is also valid with a *ctype* of STATUS. This form of the status command has the format shown below by Figure 46.

	msb							lsb
opcode	RECORDING SPEED (DB_{16})							
operand[0]	$7F_{16}$							

Figure 46 – RECORDING SPEED status command format

In the response frame generated by the Tape recorder/player subunit, the current recording speed is returned in *operand[0]*.

4.21 RECORDING TIME Command

The RECORDING TIME status command requests the Tape recorder/player subunit to return the recording time from the medium of the inserted cassette. This command uses a modified version of the consumer time code format. The format of the RECORDING TIME status command is illustrated by Figure 47.

	msb							lsb
opcode	RECORDING TIME (54_{16})							
operand[0]	71_{16}							
operand[1]	FF_{16}							
…								
operand[4]								

Figure 47 – RECORDING TIME status command format

If the Tape recorder/player subunit is able to return a STABLE response to the RECORDING TIME status command, the AV/C response frame has the format illustrated by Figure 48 below:

	msb							lsb
opcode	RECORDING TIME (54_{16})							
operand[0]	71_{16}							
operand[3]	x	x	frames (high)		frames (low nibble)			
operand[4]	x	seconds (high)			seconds (low nibble)			
operand[5]	x	minutes (high)			minutes (low nibble)			
operand[6]	x	x	hours (high)		hours (low nibble)			

Figure 48 – RECORDING TIME response format

The *frames*, *seconds*, *minutes* and *hours* operands indicate the recording time, and are specified in BCD format. The bits indicated by (x) are to be treated as don't care values when analyzing the results.

For D-VHS, the *frames* value will always be zero.

4.22 RELATIVE TIME COUNTER Command

The relative time counter is the value displayed on a VCR device as the tape medium moves forward or backward. This value is relative to any arbitrary position established by the user; pushing a reset button would normally reset the counter to zero, independent of the medium position. There is no relationship between data on the tape and the RTC counter value; the RTC is simply an indicator maintained by the Tape recorder/player subunit, based on transport motion.

The Relative Time Counter (RTC) command has three functions for the Tape recorder/player subunit, determined by the value of *ctype* and *operand[0]*. The first requests the Tape recorder/player subunit to search for a specified RTC position. The second requests the Tape recorder/player subunit to return the current RTC value. And the third sets the RTC value for the Tape recorder/player subunit.

The first format of the RELATIVE TIME COUNTER command shall have a *ctype* value of CONTROL and is illustrated below:

	msb							lsb
opcode	RELATIVE TIME COUNTER (57_{16})							
operand[0]	20_{16}							
operand[1]	sign and frame							
operand[2]	second							
operand[3]	minute							
operand[4]	hour							

Figure 49 – RELATIVE TIME COUNTER control command format

In the case of both DVCR and D-VHS VCR, the RELATIVE TIME COUNTER control command requests the Tape recorder/player subunit to be paused in playback mode immediately after searching for the specified RTC position.

The most significant bit of *sign and frame* indicates a minus or plus sign. A value of zero indicates a plus sign while a value of one indicates a minus sign. The value of the least significant seven bits of the *sign and frame* field is encoded in binary coded decimal (BCD) format where the upper three bits and lower four bits represent decimal digits. If the least significant seven bits of the *sign and frame* field or the *second* field are not used, they shall have a value of $7F_{16}$. The values *second*, *minute* and *hour* are all encoded in binary coded decimal (BCD) format where each nibble of a byte represents a decimal digit. The value -00:00:00:00 (minus zero frame, zero second, zero minute, zero hour) is not valid. The response frame format is identical to the control command frame.

NOTE – In the case of D-VHS VCR, the value of *hour* may be between zero and 99.

The second format of the RELATIVE TIME COUNTER command shall have a *ctype* value of STATUS and is illustrated by the figure below:

	msb							lsb
opcode	RELATIVE TIME COUNTER (57_{16})							
operand[0]	71_{16}							
operand[1]	FF_{16}							
operand[2]	FF_{16}							
operand[3]	FF_{16}							
operand[4]	FF_{16}							

Figure 50 – RELATIVE TIME COUNTER status command format

If the Tape recorder/player subunit is able to return a STABLE response to the RELATIVE TIME COUNTER status command, the AV/C response frame has the format illustrated below:

	msb							lsb
opcode	RELATIVE TIME COUNTER (57_{16})							
operand[0]	71_{16}							
operand[1]	sign and frame							
operand[2]	second							
operand[3]	minute							
operand[4]	hour							

Figure 51 – RELATIVE TIME COUNTER response format

The definition of the operands is the same as for the control command with operand[0] = 20_{16}.

The third command format of the RELATIVE TIME COUNTER command shall have a *ctype* value of CONTROL and is illustrated by figure below:

	msb							lsb
opcode	RELATIVE TIME COUNTER (57_{16})							
operand[0]	71_{16}							
operand[1]	sign and frame							
operand[2]	second							
operand[3]	minute							
operand[4]	hour							

Figure 52 – RELATIVE TIME COUNTER control command format

The definition of the operands is the same as for the control command with operand[0] = 20_{16}. The response frame format is identical to the command frame.

4.23 SEARCH MODE Command

The SEARCH MODE status command is used to query the current search mode and search destination, if any, of a Tape recorder/player subunit. The format of the SEARCH MODE status command is illustrated by Figure 53 below:

	msb							lsb
opcode	SEARCH MODE (50_{16})							
operand[0]	FF_{16}							
…								
operand[4]								

Figure 53 – SEARCH MODE status command format

The response frame returned after a SEARCH MODE status command differs from many others; in addition to the five operands being returned with new information, the opcode is updated to reflect the current operation in progress. The possible combinations for the return of *opcode* and the resultant operands are defined in the following table:

Table 36 – SEARCH MODE response values

Opcode	Value	operand[0]	operand[1]	operand[2]	operand[3]	operand[4]
SEARCH MODE	50_{16}	OTHER MODE (60_{16})	FF_{16}	FF_{16}	FF_{16}	FF_{16}
TIME CODE	51_{16}	SEARCHING (20_{16})	frame	second	minute	hour
ABSOLUTE TRACK NUMBER (DVCR)	52_{16}	SEARCHING (20_{16})	*ATN_data* (DVCR)			FF_{16}
ABSOLUTE TRACK NUMBER (D-VHS)	52_{16}	SEARCHING (20_{16})	*ATN_data* (D-VHS)			$0F_{16}$
FORWARD	55_{16}	*measurement_unit*	count	FF_{16}	FF_{16}	FF_{16}
BACKWARD	56_{16}	*measurement_unit*	count	FF_{16}	FF_{16}	FF_{16}
RELATIVE TIME COUNTER	57_{16}	SEARCHING (20_{16})	sign and frame	second	minute	hour
SMPTE/EBU TIME CODE	59_{16}	SEARCHING (20_{16})	frame and s1, s2 flags	second and s3 flag	minute and s4 flag	hour and s5, s6 flags

If the Tape recorder/player subunit's transport mechanism is not active in either a TIME CODE, ABSOLUTE TRACK NUMBER, FORWARD, BACKWARD or RELATIVE TIME COUNTER search, the same *opcode*, 50_{16}, is returned and the first operand indicates that the Tape recorder/player subunit is not in one of these search modes. Otherwise, the *opcode* is updated with the *opcode* used to commence the search and the operands reflect the intended destination of the search.

The TIME CODE values, *frame*, *second*, *minute* and *hour*, are all encoded in binary coded decimal (BCD) format where each nibble of a byte represents a decimal digit.

The most significant bit of *sign and frame* indicates a minus or plus sign. A value of zero indicates a plus sign, while value of one indicates a minus sign. The value of the least significant seven bits of *sign and frame* is encoded in binary coded decimal (BCD) format where the upper three bits and lower four bits represent decimal digits.

The RTC values, *second*, *minute* and *hour*, are all encoded in binary coded decimal (BCD) format where each nibble of a byte represents a decimal digit. The *ATN_data* returned is encoded in the same fashion as described in the ABSOLUTE TRACK NUMBER command. The *measurement_unit* and *count* values are as described in the BACKWARD and FORWARD commands.

The SEARCH MODE command may also be used with a *ctype* of NOTIFY. The SEARCH MODE notification service is useful to controllers who are monitoring a subunit. The actions for which SEARCH MODE provides status information can take a very long time to complete. Upon discovering that a subunit is engaged in one of these long operations, a controller can issue a SEARCH MODE notify command, so that it will be notified when the operation terminates. Note that the operation may or may not terminate successfully.

The target shall send a CHANGED response whenever the status as maintained by the SEARCH MODE command changes. The CHANGED response shall indicate the latest SEARCH status or OTHER MODE if the target has terminated the SEARCH MODE. The termination may be because the destination was found, or because it was not found and the end of the media was reached, or because it was interrupted by a higher priority event such as the user pressing the STOP button.

4.24 SMPTE/EBU RECORDING TIME Command

The SMPTE/EBU RECORDING TIME command supports the set up and query of preset values, or reading SMPTE/EBU recording time values from the medium.

The SMPTE/EBU RECORDING TIME command with *ctype* of CONTROL has one function, determined by operand [0]. It is used to preset a SMPTE/EBU recording time.

The format of the SMPTE/EBU RECORDING TIME control command frame is defined as follows:

	msb							lsb
opcode	SMPTE/EBU RECORDING TIME ($5C_{16}$)							
operand[0]	70_{16}							
operand[1]	s2	s1	tens of frames		frames			
operand[2]	s3	tens of seconds			seconds			
operand[3]	s4	tens of minutes			minutes			
operand[4]	s6	s5	tens of hours		hours			

Figure 54 – SMPTE/EBU RECORDING TIME control command format (preset value)

The *s1*, *s2*, *s3*, *s4*, *s5* and *s6* fields collectively encode SMPTE/EBU time code information. These fields represent information encoded by the table below: Refer to ANSI/SMPTE 12M-1995, and ANSI/SMPTE 262M-1995.

Table 37 – SMPTE/EBU time code information

SMPTE/EBU data	s1	s2	s3	s4	s5	s6
Vertical interval time code (VITC)	14	15	35	55	74	75
Longitudinal time code (LTC)	10	11	27	43	58	59

Each of the *frames*, *seconds*, *minutes* and *hours* fields is encoded in binary coded decimal (BCD) format where the least significant nibble and the remaining most significant bits are each a field that represents a decimal digit.

When this control command is executed, it will establish a preset SMPTE/EBU RECORDING TIME value. This value will be written to the medium when a subsequent RECORD or EDIT MODE command is executed, beginning with the first recorded frame.

The SMPTE/EBU RECORDING TIME command can also be used with a *ctype* of STATUS, and has two functions, determined by operand [0]. One is used to read the preset values of SMPTE/EBU time code, and second one is to read the current value from the medium.

The frame format of SMPTE/EBU RECORDING TIME status command to read the preset value is defined as follows:

	msb							lsb
opcode	SMPTE/EBU RECORDING TIME ($5C_{16}$)							
operand[0]	70_{16}							
operand[1]	FF_{16}							
operand[2]	FF_{16}							
operand[3]	FF_{16}							
operand[4]	FF_{16}							

Figure 55 – SMPTE/EBU RECORDING TIME status command format (read preset value)

When the subunit is able to return a stable response, the STATUS command response frame will be exactly the same as the CONTROL command frame.

The frame format of the SMPTE/EBU RECORDING TIME status command to read the current data is defined as follows:

	msb							lsb
opcode	SMPTE/EBU RECORDING TIME ($5C_{16}$)							
operand[0]	71_{16}							
operand[1]	FF_{16}							
operand[2]	FF_{16}							
operand[3]	FF_{16}							
operand[4]	FF_{16}							

Figure 56 – SMPTE/EBU RECORDING TIME status command format (read current value)

The contents of the stable response frame for a data mode status command will vary, depending on the format of the recorded medium - either consumer or SMPTE/EBU time code. The general frame structure will be as follows:

	msb							lsb
opcode	SMPTE/EBU RECORDING TIME (opcode $5C_{16}$ or 54_{16} - see note)							
operand[0]	71_{16}							
operand[1]	time code data (either SMPTE/EBU or consumer format)							
operand[2]								
operand[3]								
operand[4]								

Figure 57 – SMPTE/EBU RECORDING TIME response format (read current value)

Note that the opcode returned in the response frame can be either $5C_{16}$ (SMPTE/EBU RECORDING TIME command) or 54_{16} (RECORDING TIME - for consumer format). The following table illustrates the rules for determining what the return value will be:

Table 38 – Opcode and data in the response frame

when the medium is recorded in this format	the opcode in the response frame will be	the format of the time code data will be
SMPTE/EBU	$5C_{16}$	SMPTE/EBU
consumer	54_{16}	consumer

The SMPTE/EBU RECORDING TIME command can also be used with a ctype of NOTIFY. This means that a controller will be notified when the preset value is changed. The format of the NOTIFY command is the same as the one of the STATUS command to read preset value. The response frame is the same as the STATUS, preset response frame.

The following table helps to clarify the valid combinations of command ctype and operation mode for both the SMPTE/EBU RECORDING TIME and the regular RECORDING TIME commands:

Table 39 – Valid combinations of ctype and operation mode

		RECORDING TIME	SMPTE/EBU RECORDING TIME
ctype	mode	valid combination?	valid combination?
CONTROL	search	X	X
	preset	X	YES
	data	X	X
STATUS	search	X	X
	preset	X	YES
	data	YES	YES
SPECIFIC INQUIRY	search	X	X
	preset	X	YES
	data	X	X
NOTIFY	search	X	X
	preset	X	YES
	data	X	X
GENERAL INQUIRY	N/A	YES	YES

4.25 SMPTE/EBU TIME CODE Command

The SMPTE/EBU TIME CODE command with a *ctype* value of CONTROL has two functions determined by the value of *operand[0]*. The first requests the Tape recorder/player subunit to search for a specified SMPTE/EBU time code on the medium, and the second presets a SMPTE/EBU time code value.

The format of the SMPTE/EBU TIME CODE control command frame to search for a specified SMPTE/EBU time code is as follows:

	msb							lsb
opcode	SMPTE/EBU TIME CODE (59_{16})							
operand[0]	20_{16}							
operand[3]	s2	s1	tens of frames			units of frames		
operand[4]	s3	tens of seconds				units of seconds		
operand[5]	s4	tens of minutes				units of minutes		
operand[6]	s6	s5	tens of hours			units of hours		

Figure 58 – SMPTE/EBU TIME CODE control command format (search value)

All of the operands are as described in the SMPTE/EBU RECORDING TIME command description. The response frame format is identical to the control command frame.

The frame format of SMPTE/EBU TIME CODE control command to preset the SMPTE/EBU time code is the same as for the search mode, but operand[0] has the value 70_{16}.

When this control command is executed, it will establish a preset time code value. This value will be written to the medium when a subsequent RECORD command is executed, beginning with the first recorded frame.

The SMPTE/EBU TIME CODE command is also used with a *ctype* of STATUS, and has two functions determined by operand [0]: to read the preset time code, and to read the current time code from the medium.

The format of the SMPTE/EBU TIME CODE status command frame to read the preset SMPTE/EBU time code is defined as follows:

	msb							lsb
opcode	SMPTE/EBU TIME CODE (59_{16})							
operand[0]	70_{16}							
operand[3]	FF_{16}							
operand[4]	FF_{16}							
operand[5]	FF_{16}							
operand[6]	FF_{16}							

Figure 59 – SMPTE/EBU TIME CODE status command format (read preset value)

When the subunit is able to return a stable response, the STATUS command response frame will be the same as the CONTROL command frame.

The frame format of SMPTE/EBU TIME CODE status command to read the current SMPTE/EBU time code from the medium is the same as the frame above, but operand[0] will have the value 71_{16}.

It is important to understand the relationship between the two time code commands (TIME CODE and SMPTE/EBU TIME CODE), and the two time code formats that are defined (consumer and SMPTE/EBU time code). The results of a status command will depend on which command is issued, and the format of the medium being investigated.

The general format of the response will be as follows:

	msb							lsb
opcode	SMPTE/EBU TIME CODE (opcode 59_{16} OR 51_{16} - see note)							
operand[0]	71_{16}							
operand[3]	time code data (either SMPTE/EBU or consumer format)							
operand[4]								
operand[5]								
operand[6]								

Figure 60 – SMPTE/EBU TIME CODE response format (read current value)

The following table illustrates the rules for determining what the return value will be:

Table 40 – Opcode and data in the response frame

when the medium is recorded in this format...	...and this opcode is used for the STATUS command...	...then this will be the RESPONSE opcode...	...and this will be the data that is returned
SMPTE/EBU	51_{16}	51_{16}	consumer
	59_{16}	59_{16}	SMPTE/EBU
consumer	51_{16}	51_{16}	consumer
	59_{16}	51_{16}	consumer

There is one additional special case for the response frame from the STATUS, data command. Under normal circumstances, the mode value in operand[0] will be 71_{16} upon return. However, if the subunit is unable to reliably read the data, it may estimate the data. If it does this, then it shall return a value of 61_{16} in operand[0].

Example: If there is a section of DVCR tape which does not have time code on it, but the VCR was able to read time code before that section, then the VCR may estimate the time code in the non-time-coded area. It shall return 61_{16} and the estimated time code data.

The SMPTE/EBU TIME CODE command can also be used with a ctype of SPECIFIC INQUIRY, which has two functions, determined by operand[0]. The SPECIFIC INQUIRY command with operand [0] of value 70_{16} inquires about support for establishing a preset SMPTE/EBU time code, and operand[0] with value 20_{16} inquires about support for searching for the SMPTE/EBU time code value. As defined in the AV/C specification, the frames for SPECIFIC INQUIRY commands have the same format as their corresponding CONTROL commands.

The SMPTE/EBU TIME CODE command can also be used with a ctype of NOTIFY, to monitor the preset value of SMPTE/EBU time code. This means that a controller will be notified when the preset value is changed. The formats of the NOTIFY command and its response are the same as the STATUS command to preset the SMPTE/EBU time code.

The following table illustrates the valid and invalid combinations of command ctype and operation mode for the TIME CODE and SMPTE/EBU TIME CODE commands:

Table 41 – Valid combinations of ctype and operation mode

ctype	mode	TIME CODE	SMPTE/EBU TIME CODE
		valid combination?	valid combination?
CONTROL	search	YES	YES
	preset	X	YES
	data	X	X
STATUS	search	X	X
	preset	X	YES
	data	YES	YES
SPECIFIC INQUIRY	search	X	YES
	preset	X	YES
	data	X	X
NOTIFY	search	X	X
	preset	X	YES
	data	X	X
GENERAL INQUIRY	N/A	YES	YES

4.26 TAPE PLAYBACK FORMAT Command

The TAPE PLAYBACK FORMAT control command specifies the digital playback format of the Tape recorder/player subunit. The support levels for this command will depend on the type of Tape recorder/player subunit, as defined below:

Table 42 – Support level of TAPE PLAYBACK FORMAT Command

Type of Tape recorder/player subunit	Support level (by *ctype*)			Comments
	C	S	N	
DVCR	–	–	–	
D-VHS	O	M	–	
8mm	–	–	–	

The format of the TAPE PLAYBACK FORMAT control command is illustrated by the figure below:

	msb							lsb
opcode	TAPE PLAYBACK FORMAT ($D3_{16}$)							
operand[0]	medium_type							
operand[1]	format_parameter							
...								
operand[8]								

Figure 61 – TAPE PLAYBACK FORMAT control command format

The ***medium_type*** indicates the kind of the medium used in the Tape recorder/player subunit and the format of the ***format_parameter***. A *medium_type* value of 00001_2 represents the medium used in the D-VHS VCR while a medium_type value of 11111_2 represents the medium used in the DVCR, as defined in ABSOLUTE TRACK NUMBER command.

The format of the ***format_parameter*** for D-VHS VCR, illustrated by figure below, shall have the following parameters: ***af[n](n=0..9), ECC_block_size, ECC_block_number, Program_mode, Scanner_rotation_speed, 1.001_flag, Outer_interleave, Recording_mode, Application_ID, Detail_information_of_application_ID, Time_compression_ratio, extension_code_A, extension_code_B,*** and ***Reserved*** field.

	msb							lsb
opcode	TAPE PLAYBACK FORMAT ($D3_{16}$)							
operand[0]	00001_2					af[0]	af[1]	af[2]
operand[1]	af[3]	af[4]	af[5]	af[6]	af[7]	af[8]	af[9]	extension_code_A
operand[2]	extension_code_B							
operand[3]	ECC_block_size	ECC_block_number		Program_mode			Scanner_rotation_speed	
operand[4]	1.001_flag	Outer_interleave			Recording_mode			
operand[5]	Reserved							
operand[6]	Application_ID					Reserved		
operand[7]	Detail_information_of_application_ID				Time_compression_ratio			
operand[8]	Reserved							

Figure 62 – TAPE PLAYBACK FORMAT control command format for D-VHS VCR

The ***af[0]*** bit indicates whether or not the ***ECC_block_size*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[1]*** bit indicates whether or not the ***ECC_block_number*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[2]*** bit indicates whether or not the ***Program_mode*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[3]*** bit indicates whether or not the ***Scanner_rotation_speed*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[4]*** bit indicates whether or not the ***1.001_flag*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[5]*** bit indicates whether or not the ***Outer_interleave*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[6]*** bit indicates whether or not the ***Recording_mode*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[7]*** bit indicates whether or not the ***Application_ID*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[8]*** bit indicates whether or not the ***Detail_information_of_application_ID*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***af[9]*** bit indicates whether or not the ***Time_compression_ratio*** field is available. A value of zero indicates unavailable while a value of one indicates available.

The ***ECC_block_size*** field indicates the size of data, which constitute the product codes of main code. A value of zero is defined that one ECC block consists of 112 sync blocks (which includes data sync blocks with inner parity, and outer parity sync blocks).

The ***ECC_block_number*** field indicates the number of ECC blocks per track. A value of 00_2 is defined as three ECC blocks per track. This parameter is prepared to add more definition for using a different number of ECC blocks per track in the other D-VHS mode to be determined as well as the *ECC_block_size*.

The ***Program_mode*** field indicates the structure of main code area. A value of 000_2 is defined as one program per track. This parameter is prepared to add more definition for using the multiple program per track in the other D-VHS mode to be determined as well as the *ECC_block_size*.

The ***Scanner_rotation_speed*** field indicates the scanner rotation times per minute. A value of 00_2 is defined as 1800 rpm. This parameter is prepared to add more definition for such as the high-speed data transfer, which is necessary to raise the scanner rotation speed in the other D-VHS mode to be determined.

The ***1.001_flag*** field indicates whether or not the scanner rotation speed is 59.94Hz. A value of zero indicates that an actual scanner rotation speed is identical with the value specified by the *Scanner_rotation_speed* . While a value of one indicates that an actual scanner rotation speed is identical with 1/1.001 of the value specified by the *Scanner_rotation_speed* for a synchronized recording or playback by external 59.94Hz NTSC sync signal. Both values are valid.

The ***Outer_interleave*** field indicates the number of tracks, which constitute on interleave block for the outer error correcting codes. A value of 000_2 is defined as 6 tracks. This parameter is prepared to add more definition for an another interleave block in the other D-VHS mode to be determined.

The ***Recording_mode*** field indicates the current recording mode of the D-VHS VCR such as STD, High-Speed (HS) and Low-Speed (LS) modes, and specifies the appropriate mode for the application to be used. This parameter is prepared to add more definition for another recording mode.

Six values for *recording_mode* are specified by the D-VHS specifications, as shown in the table below and all other values shall be reserved:

Table 43 – Recording_mode for DVHS

Recording_mode for D-VHS	Value	Comment
STD	0000_2	
HS	0001_2	
LS2	0010_2	
LS3	0011_2	
LS5	0101_2	
LS7	0111_2	

– HS : This mode can provide up to twice the data rate of the STD mode.

– LS2 : This mode can provide up to 1/2 the data rate of the STD mode.

– LS3 : This mode can provide up to 1/3 the data rate of the STD mode.

– LS5 : This mode can provide up to 1/5 the data rate of the STD mode.

– LS7 : This mode can provide up to 1/7 the data rate of the STD mode.

The ***Application_ID*** field provides the definitions of Subcode data area and Main data area. This parameter specifies the logical format of the medium. A value of 00000_2 is defined as MPEG2, which indicates the format for MPEG2 signal.

The ***Detail_information_of_application_ID*** field indicates the kind of contents recorded in Main code area. A value of 0000_2 is defined as MPEG2 transport stream.

The ***Time_compression_ratio*** field indicates the time compression ratio of the recorded data. A value of 0000_2 is defined as 1:1, and indicates that the recorded data is not time-compressed. This parameter is prepared to add more definition for recording the time-compressed data in the other D-VHS mode to be determined.

The values in the ***extension_code_A*** field and the ***extension_code_B*** field shall be zero, and all other values are reserved.

The ***Reserved*** field shall be filled with zero.

The TAPE PLAYBACK FORMAT command may also be used with a *ctype* value of STATUS. The format of this command is illustrated below:

	msb							lsb
opcode	TAPE PLAYBACK FORMAT ($D3_{16}$)							
operand[0]	FF_{16}							
…								
operand[8]								

Figure 63 – TAPE PLAYBACK FORMAT status command format

If the Tape recorder/player subunit is able to return a STABLE response to the TAPE PLAYBACK FORMAT status command, the AV/C response frame has the format illustrated below:

	msb							lsb
opcode	TAPE PLAYBACK FORMAT ($D3_{16}$)							
operand[0]	medium_type							
operand[1]	format_parameter							
...								
operand[8]								

Figure 64 – TAPE PLAYBACK FORMAT response format

In the case of D-VHS VCR, the AV/C response frame has the format illustrated by the following diagram:

	msb							lsb
opcode	TAPE PLAYBACK FORMAT ($D3_{16}$)							
operand[0]	00001_2					af[0]	af[1]	af[2]
operand[1]	af[3]	af[4]	af[5]	af[6]	af[7]	af[8]	af[9]	extension_code_A
operand[2]	extension_code_B							
operand[3]	ECC_block_size	ECC_block_number		Program_mode			Scanner_rotation_speed	
operand[4]	1.001_flag	Outer_interleave			Recording_mode			
operand[5]	Reserved							
operand[6]	Application_ID					Reserved		
operand[7]	Detail_information_of_application_ID				Time_compression_ratio			
operand[8]	Reserved							

Figure 65 – TAPE PLAYBACK FORMAT response format for D-VHS VCR

The meanings of the fields, af[n](n=0..9), extension_code_A, extension_code_B, ECC_block_size, ECC_block_number, Program_mode, Scanner_rotation_speed, 1.001_flag, Outer_interleave, Recording_mode, Application_ID, Detail_information_of_application_ID, Time_compression_ratio and Reserved are as previously described in this TAPE PLAYBACK FORMAT control command.

4.27 TAPE RECORDING FORMAT Command

The TAPE RECORDING FORMAT control command specifies the recording format of the Tape recorder/player subunit. The support levels for this command will depend on the type of Tape recorder/player subunit, as defined below:

Table 44 – Support level of TAPE RECORDING FORMAT Command

Type of Tape recorder/player subunit	Support level (by *ctype*) C	S	N	Comments
DVCR	–	–	–	
D-VHS	O	M	–	
8mm	–	–	–	

The format of the TAPE RECORDING FORMAT control command is illustrated by the figure below:

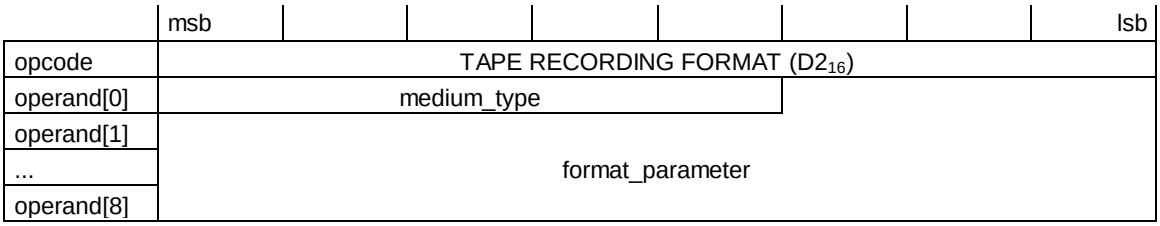

Figure 66 – TAPE RECORDING FORMAT control command format

The meanings of the operands *medium_type* and *format_parameter* are as previously described in the TAPE PLAYBACK FORMAT command.

The format of the ***format_parameter*** for D-VHS VCR, illustrated below, shall have the following parameters: ***af[n](n=0..9), ECC_block_size, ECC_block_number, Program_mode, Scanner_rotation_speed, 1.001_flag, Outer_interleave, Recording_mode, Application_ID, Detail_information_of_application_ID, Time_compression_ratio, extension_code_A, extension_code_B,*** and ***Reserved*** field.

	msb							lsb
opcode	TAPE RECORDING FORMAT ($D2_{16}$)							
operand[0]	00001_2					af[0]	af[1]	af[2]
operand[1]	af[3]	af[4]	af[5]	af[6]	af[7]	af[8]	af[9]	*extension_code_A*
operand[2]	*extension_code_B*							
operand[3]	*ECC_block_size*	*ECC_block_number*		*Program_mode*			*Scanner_rotation_speed*	
operand[4]	*1.001_flag*	*Outer_interleave*			*Recording_mode*			
operand[5]	Reserved							
operand[6]	*Application_ID*					Reserved		
operand[7]	*Detail_information_of_application_ID*				*Time_compression_ratio*			
operand[8]	Reserved							

Figure 67 – TAPE RECORDING FORMAT control command format for D-VHS VCR

The meanings of the fields, af[n](n=0..9), extension_code_A, extension_code_B, ECC_block_size, ECC_block_number, Program_mode, Scanner_rotation_speed, 1.001_flag, Outer_interleave, Recording_mode, Application_ID, Detail_information_of_application_ID, Time_compression_ratio and Reserved are as previously described in the TAPE PLAYBACK FORMAT command.

The default recording mode for D-VHS VCR corresponds to a zero value for all of the parameters in this command. So, a recording can be started without first sending a TAPE RECORDING FORMAT command.

The TAPE RECORDING FORMAT command may also be used with a ctype value of STATUS. The format of this status command is illustrated by the figure below:

	msb							lsb
opcode	TAPE RECORDING FORMAT ($D2_{16}$)							
operand[0]	FF_{16}							
…								
operand[8]								

Figure 68 – TAPE RECORDING FORMAT status command format

If the Tape recorder/player subunit is able to return a STABLE response to the TAPE RECORDING FORMAT status command, the AV/C response frame has the format illustrated by the figure below:

	msb							lsb
opcode	TAPE RECORDING FORMAT ($D2_{16}$)							
operand[0]	medium_type							
operand[1]	format_parameter							
...								
operand[8]								

Figure 69 – TAPE RECORDING FORMAT response format

In the case of D-VHS VCR, the AV/C response frame has the format illustrated by the figure below:

	msb							lsb
opcode	TAPE RECORDING FORMAT ($D2_{16}$)							
operand[0]	00001_2					af[0]	af[1]	af[2]
operand[1]	af[3]	af[4]	af[5]	af[6]	af[7]	af[8]	af[9]	extension_code_A
operand[2]	extension_code_B							
operand[3]	ECC_block_size	ECC_block_number		Program_mode			Scanner_rotation_speed	
operand[4]	1.001_flag	Outer_interleave			Recording_mode			
operand[5]	Reserved							
operand[6]	Application_ID					Reserved		
operand[7]	Detail_information_of_application_ID				Time_compression_ratio			
operand[8]	Reserved							

Figure 70 – TAPE RECORDING FORMAT response format for D-VHS VCR

The fields, af[n](n=0..9), extension_code_A, extension_code_B, ECC_block_size, ECC_block_number, Program_mode, 1.001_flag, Outer_interleave, Recording_mode, Application_ID, Detail_information_of_application_ID, Time_compression_ratio, and Reserved, are used as previously described for the TAPE RECORDING FORMAT control command.

4.28 TIME CODE Command

The TIME CODE command has two functions determined by the value of *ctype* and *operand[0]*. The first requests the Tape recorder/player subunit to search for a specified time code on the medium while the second requests the Tape recorder/player subunit to return the time code value for the current medium position. The support levels for this command will depend on the type of Tape recorder/player subunit, as defined below:

Table 45 – Support level of TAPE RECORDING FORMAT Command

Type of Tape recorder/player subunit	Support level (by *ctype*)			Comments
	C	S	N	
DVCR	R	M	–	
D-VHS	O	O	–	
8mm	O	O	–	

The first format of the TIME CODE command shall have a ctype value of CONTROL and is illustrated by Figure below:

	msb							lsb
opcode	TIME CODE (51_{16})							
operand[0]	20_{16}							
operand[1]	frame							
operand[2]	second							
operand[3]	minute							
operand[4]	hour							

Figure 71 – TIME CODE control command format

In the case of DVCR, the TIME CODE control command requests the Tape recorder/player subunit to be paused in playback mode immediately after searching for the specified time code on the medium.

The values *frame*, *second*, *minute* and *hour* are all encoded in binary coded decimal (BCD) format where each nibble of a byte represents a decimal digit. The response frame format is identical to the command frame.

If the target does not have the ability to support frames, then it may ignore the value passed in for the *frame* operand and accept the command.

The second format of the TIME CODE command, reading the time code from the medium, shall have a *ctype* value of STATUS and is illustrated by Figure 72 below:

	msb							lsb
opcode	TIME CODE (51_{16})							
operand[0]	71_{16}							
operand[1]	FF_{16}							
…								
operand[4]								

Figure 72 – TIME CODE status command format

If the Tape recorder/player subunit is able to return a STABLE response to the TIME CODE status command then the AV/C response frame has the format illustrated by Figure 73 below:

	msb							lsb
opcode	TIME CODE (51_{16})							
operand[0]	71_{16}							
operand[1]	frame							
operand[2]	second							
operand[3]	minute							
operand[4]	hour							

Figure 73 – TIME CODE response format

The values *frame*, *second*, *minute* and *hour* are all encoded in binary coded decimal (BCD) format where each nibble of a byte represents a decimal digit.

If the target does not support frames, then it shall return a value of $7F_{16}$ for the *frame* operand.

4.29 TRANSPORT STATE Command

The TRANSPORT STATE status or notify command is used to inquire as to the current state of the medium in the transport mechanism. The format of the TRANSPORT STATE command is illustrated by Figure 74 below:

	msb							lsb
opcode	TRANSPORT STATE ($D0_{16}$)							
operand[0]	$7F_{16}$							

Figure 74 – TRANSPORT STATE command format

The TRANSPORT STATE command shall be used only with a *ctype* of STATUS or NOTIFY. The information returned is formatted in a response frame, as shown by Figure 75 below:

	msb							lsb
opcode	transport_mode							
operand[0]	transport_state							

Figure 75 – TRANSPORT STATE response format

In the response frame returned after a TRANSPORT STATE status or notify command, the ***transport_mode*** indicates the operating mode of the transport mechanism. The transport state is represented by previously defined opcodes, LOAD MEDIUM, RECORD, PLAY and WIND. Within each category of *transport_mode*, the value returned for *operand[0]* represents the transport state, as defined by Table 46 below:

Table 46 – Transport state values

Transport mode	Transport state	Value	Comment
LOAD MEDIUM ($C1_{16}$)	EJECT	60_{16}	No medium is present
RECORD ($C2_{16}$)	As defined by Table 29		Any of the values defined for the RECORD command may be returned to indicate the corresponding transport state
	UNSPECIFIED INSERT	30_{16}	One of the insert states, 31_{16} - 34_{16}, 36_{16} - 38_{16}
	UNSPECIFIED INSERT PAUSE	40_{16}	One of the paused insert states, 41_{16} - 44_{16}, 46_{16} - 48_{16}
PLAY ($C3_{16}$)	As defined by Table 27		Any of the values defined for the PLAY command, except 30_{16} and 40_{16}, may be returned to indicate the corresponding transport state
	UNSPECIFIED SLOW FORWARD	31_{16}	One of the SLOW FORWARD trick play states, 31_{16} - 37_{16}
	UNSPECIFIED FAST FORWARD	$3F_{16}$	One of the FAST FORWARD trick play states, 39_{16} - $3F_{16}$
	UNSPECIFIED SLOW REVERSE	41_{16}	One of the SLOW REVERSE trick play states, 41_{16} - 47_{16}
	UNSPECIFIED FAST REVERSE	$4F_{16}$	One of the FAST REVERSE trick play states, 49_{16} - $4F_{16}$
WIND ($C4_{16}$)	As defined by Table 47		Any of the values defined for the WIND command may be returned to indicate the corresponding transport state
	STOP EMERGENCY	30_{16}	Unexpected conditions have forced a stop
	STOP CONDENSATION(DEW)	31_{16}	Possible risk of transport damage

If a Tape recorder/player subunit is unable to precisely determine its transport state, it may return a response that indicates one of the UNSPECIFIED states described above.

A Tape recorder/player subunit that supports only one speed for any of the trick play groups, slow forward, fast forward, slow reverse, or fast reverse, shall return the corresponding UNSPECIFIED code when in one of the trick play states.

A Tape recorder/player subunit that supports two or more speeds for any of the trick play groups shall return a code that corresponds to the actual speed of the transport mechanism, as defined by the mapping described in 4.14 but it may not be equal to the *playback_mode* operand of the PLAY control command that initiated the operation.

NOTE – The value returned in the response is not required to be equal to the value used to activate the transport mechanism. For example, a VCR that mapped 1/30 playback speed to SLOWEST, SLOW FORWARD 6 and SLOW FORWARD 5 might have playback initiated by a PLAY control command that specified SLOW FORWARD 6. A subsequent TRANSPORT STATE status could return SLOW FORWARD 5 to accurately reflect the actual transport speed.

4.30 WIND Command

The WIND control command is used to control movement of the medium in the transport mechanism. Motion initiated by the WIND control command differs from motion initiated by the PLAY or RECORD control commands in that it may be unmonitored with respect to precise track position. The format of the WIND control command is illustrated by Figure 76 below.

	msb							lsb
opcode	WIND ($C4_{16}$)							
operand[0]	subfunction							

Figure 76 – WIND control command format

All of the WIND subfunctions typically cause the transport mechanism's playback and record head(s) to be unloaded from the medium and then cause the motion indicated. The value of the *subfunction* operand determines the operation performed by the target, as defined by the table below:

Table 47 – WIND subfunctions

Subfunction	Value	Support level	Action
HIGH SPEED REWIND	45_{16}	O	Move the medium toward the beginning of medium as quickly as possible
STOP	60_{16}	M	Halt all transport mechanism motion
REWIND	65_{16}	M	Move the medium toward the beginning of medium
FAST FORWARD	75_{16}	M	Move the medium away from the beginning of medium

If no cassette is loaded, the Tape recorder/player subunit shall refuse the WIND control command and return a *response* of REJECTED.

4.31 WRITE MIC Command

The WRITE MIC control command is used to store variable-length data in cassette nonvolatile memory (MIC). A controller is expected to have issued an OPEN MIC control command to the target subunit before performing any MIC read or write operations. The format of the WRITE MIC control command is illustrated by Figure 77 below:

	msb							lsb
opcode	WRITE MIC (62_{16})							
operand[0]	data_length							
operand[1]	(most significant byte) MIC_address							
operand[2]	(least significant byte)							
operand[3]	data							
...								
operand[*n*]								

Figure 77 – WRITE MIC control command format

The *data_length* field specifies the number of bytes to be written to the MIC.

The *MIC_address* field specifies the starting offset of the data within the MIC.

– If *data_length* specifies more bytes than can be accepted by the Tape recorder/player subunit in a single operation or if the combination of *data_length* and *MIC_address* reference unimplemented memory, the Tape recorder/player subunit shall reject the WRITE MIC control command.

– If an ACCEPTED, INTERIM or REJECTED response frame is returned by the target after a WRITE MIC control command, the response frame confirms the *data_length* and the *MIC_address* but does not echo any of the data bytes written to nonvolatile memory.

In addition to the use of WRITE MIC when *ctype* has a value of CONTROL, WRITE MIC may also be used to query the Tape recorder/player subunit's capabilities to accept MIC data. The format shown by Figure 78 below, when *ctype* has a value of STATUS, is used to determine the maximum length of data that a Tape recorder/player subunit can write to MIC in a single operation:

	msb							lsb
opcode	WRITE MIC (62_{16})							
operand[0]	FF_{16}							
...								
operand[2]								

Figure 78 – WRITE MIC status command format

In the response frame returned by the Tape recorder/player subunit, *operand[0]* is replaced with the maximum data length that may be written to MIC in a single WRITE MIC operation.

5 Tape recorder/player subunit responses (informative)

This section describes expected behaviors for Tape recorder/player subunit commands which affect or query the transport state of the VCR, the LOAD MEDIUM, PLAY, RECORD and WIND control commands and the TRANSPORT STATE status and notify commands. The goal is to promote interoperability of different vendors' products.

In addition to the behaviors defined for each of these control commands, it is expected that a Tape recorder/player subunit return a REJECTED response under the following circumstances:

- The Tape recorder/player subunit or transport mechanism is in an emergency condition.
- The loaded medium is positioned at end of medium and a control command that specifies forward motion is received.
- The Tape recorder/player subunit is not powered (although the AV unit which contains the Tape recorder/player subunit is powered).
- The Tape recorder/player subunit (or the AV unit which contains it) is in a timer recording mode.
- The Tape recorder/player subunit is configured for a different device mode that prevents the operation of the control command, e.g., a camcorder which is in camera, rather than VCR, mode; or
- Vendor dependent cases that inhibit the control command, e.g., the transition to or from a RECORD mode from any transport state except stopped.

This is not an exhaustive list of conditions that cause the rejection of a LOAD MEDIUM, PLAY, RECORD or WIND control command.

Although the TRANSPORT STATE status and notify commands cannot affect the state of the transport mechanism within the Tape recorder/player subunit, it is also expected that a REJECTED response be returned after a TRANSPORT STATE command if the Tape recorder/player subunit is not powered (although the AV unit which contains the Tape recorder/player subunit is powered).

Annex A
(normative)

AV/C Tape recorder/player subunit commands in numerical order

The table below lists the AV/C Tape recorder/player commands, in numerical order by *opcode*:

Table A-1 – Tape recorder/player subunit commands

Value	Opcode	Support level (by *ctype*)		
		C	S	N
40_{16}	EDIT MODE	O	O	–
45_{16}	PRESET	O	O	–
50_{16}	SEARCH MODE	–	R	O
51_{16}	TIME CODE	*	*	–
52_{16}	ABSOLUTE TRACK NUMBER	*	*	–
53_{16}	RECORDING DATE	O	O	–
54_{16}	RECORDING TIME	–	O	–
55_{16}	FORWARD	R	–	–
56_{16}	BACKWARD	R	–	–
57_{16}	RELATIVE TIME COUNTER	R	R	–
59_{16}	SMPTE/EBU TIME CODE	O	O	O
$5A_{16}$	BINARY GROUP	O	O	O
$5C_{16}$	SMPTE/EBU RECORDING TIME	O	O	O
60_{16}	OPEN MIC	*	R	–
61_{16}	READ MIC	R	–	–
62_{16}	WRITE MIC	O	O	–
70_{16}	ANALOG AUDIO OUTPUT MODE	O	O	–
71_{16}	AUDIO MODE	O	O	–
72_{16}	AREA MODE	O	O	–
78_{16}	OUTPUT SIGNAL MODE	O	M	O
79_{16}	INPUT SIGNAL MODE	O	M	O
$C1_{16}$	LOAD MEDIUM	O	–	–
$C2_{16}$	RECORD	*	–	–
$C3_{16}$	PLAY	*	–	–
$C4_{16}$	WIND	*	–	–
CA_{16}	MARKER	R	R	O
$D0_{16}$	TRANSPORT STATE	–	M	O
$D2_{16}$	TAPE RECORDING FORMAT	*	*	–
$D3_{16}$	TAPE PLAYBACK FORMAT	*	*	–
DA_{16}	MEDIUM INFO	–	R	–
DB_{16}	RECORDING SPEED	O	O	–

In the preceding table, an asterisk in the support level column indicates that the command operands or the type of subunit determines whether the command is mandatory (M), recommended (R) or optional (O).