AV/C Digital Interface Command Set for Secure Bus System

Version 1.0
January 26, 1999

Sponsored by:
Audio/Video Working Group of the 1394 Trade Association

Approved for Release by:
1394 Trade Association Steering Committee

Abstract: This specification defines a command set for consumer and professional audio/video equipment over IEEE Std. 1394-1995. The command set makes use of the Function Control Protocol (FCP) defined by IEC-61883, Digital Interface for Consumer Electronic Audio/Video Equipment, for the transport of audio/video 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
Regency Plaza Suite 350, 2350 Mission College Blvd., Santa Clara, CA 95054, USA
http://www.1394TA.org
Copyright © 1996-1997 by the 1394 Trade Association. Permission is granted to members of the 1394 Trade Association to reproduce this document for their own use or the use of other 1394 Trade Association members only, provided this notice is included. All other rights reserved. Duplication for sale, or for commercial or for-profit use is strictly prohibited without the prior written consent of the 1394 Trade Association.

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 approved 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
Regency Plaza Suite 350
2350 Mission College Blvd.
Santa Clara, Calif. 95054

1394 Trade Association Specifications are adopted by the 1394 Trade Association without regard to patents which may exist on articles, materials or processes, or to other proprietary intellectual property which may exist within a specification. Adoption of a specification by the 1394 Trade Association does not assume any liability to any patent owner or any obligation whatsoever to those parties who rely on the specification documents. Readers of this document are advised to make an independent determination regarding the existence of intellectual property rights which may be infringed by conformance to this specification.

Copyright © 1999, 1394 Trade Association. All rights reserved.

Table of Contents

List of Figures

List of Tables

Copyright © 1999, 1394 Trade Association. All rights reserved.

Preface

This document specifies the command for secure bus system.
This command can be used for various categories relevant to the security such as Authentication and Key exchange.
One category is assigned for the Authentication and Key Exchange defined by Digital Transmission Licensing Administrator (DTLA) for their Digital Transmission Content Protection (DTCP) system.
The frame format of the command for DTCP system is also specified.
Vendor dependent category is also defined.

Copyright © 1999, 1394 Trade Association. All rights reserved.

1. References

1.1 Related Specifications

AV/C Digital Interface Command Set General Specification, version 3.0

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

IEC-61883, Digital Interface for Consumer Electronic Audio/Video Equipment

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

5C Digital Transmission Content Protection White Paper, available from Digital Transmission Licensing Administrator (DTLA)

1.2 Contact Information

1394 Trade Association (1394TA)
Home Page: http://www.1394ta.org/
Regency Plaza Suite 350
2350 Mission College Blvd.
Santa Clara, Calif. 95054, USA

Digital Transmission Licensing Administrator (DTLA)
Home Page: http://www.dtcp.com/
Email: info-request@dtcp.com

Copyright © 1999, 1394 Trade Association. All rights reserved.

2. Command set for Secure Bus System

2.1 SECURITY command

In a secure bus system, isochronous data is encrypted before transmission by the source device, and decrypted by the sink device after receiving the data. The KEY for data encryption should be transmitted from the source device to the sink device safely. This procedure is called Authentication and Key exchange. The SECURITY command is intended for content protection purposes including Authentication and Key exchange.

The general format of SECURITY command is as follows:

	msb							lsb
opcode	SECURITY ($0F_{16}$)							
operand[0]	category				(msb)			
operand[1]	category dependent field							
:								
operand[X]								(lsb)

Figure 1.1 SECURITY command frame

The value of **opcode** is $0F_{16}$. (Common Unit and Subunit command)
The *category* field specifies which variation of the SECURITY command is being issued. It can take one of the following values:

Value	category
0_{16}	Authentication and Key exchange defined by DTLA
1_{16} - D_{16}	Reserved for future specification
E_{16}	Vendor dependent security command
F_{16}	Extension for *category* field

Table 1.1 definition of category field

The value 0 of *category* field specifies that this command is used to support the DTCP Authentication and Key Exchange protocols. This command is called the **AKE command** after this.
The value E_{16} of *category* field specifies that this command is used by companies to specify their own security commands.

The format of *category dependent field* depends on the value of the *category* field.

Copyright © 1999, 1394 Trade Association. All rights reserved.

2.1.1 AKE command (informative)

This section introduces the outline of the usage of the AKE command.
The detailed specification of the usage of the AKE command is defined by DTLA.
The destination of this command is the unit of the target device.
Therefore the 5 bit *subunit_type* field of an AV/C command/response frame is equal to 11111_2 and the 3 bit *subunit_ID* field of the frame is equal to 111_2.

2.1.1.1 AKE control command frame

The AKE control command is used to exchange the messages to implement the Authentication and Key Exchange protocols. The format of this command is shown below:

	msb							lsb
opcode	$0F_{16}$							
operand[0]	category = 0000_2				AKE_ID			
operand[1]	(msb)							
operand[2]	AKE_ID dependent field							
operand[3]								
operand[4]								(lsb)
operand[5]	AKE_label							
operand[6]	number (option)				status			
operand[7]	blocks_remaining							(msb)
operand[8]	data_length							(lsb)
operand[9]	data							
:								
operand[8+ Data_length]								

Figure 1.2 AKE control command frame

AKE control commands are used to send the information used for the authentication and key exchange procedure being performed between the source and sink device. This information is sent in the *data* field and is called AKE Info.

Both the AKE Command and Response frame have the same opcode and 9 operands (operand[0-8]).
The value of each field in the response frame is identical to that of the command frame except for the *status* and *data* fields.

- If a given command frame includes *data* field, the corresponding response frame does not have a *data* field.

AKE_ID field specifies the format of the *AKE_ID dependent field*.

Currently only the encoding *AKE_ID* = 0 is defined. The other values, from 1_{16} to F_{16} are reserved for future definition.

In case of ***AKE_ID* = 0**, ***AKE_ID dependent field*** contains such information as described below.

Copyright © 1999, 1394 Trade Association. All rights reserved.

- Function code which specifies the operation of the control command. AKE control command is subdivided with this code. For example, exchange data for authentication, deliver key for decryption, and so on.
- Algorithm of AKE protocol which is used in the current AKE procedure.

AKE_label field is a unique tag which is used to distinguish a sequence of AKE commands associated with a given authentication process. The initiator of an authentication procedure can select an arbitrary value for the *AKE_label*.

The value selected should be different from other *AKE_label* values that are currently in use by the device initiating the authentication.
The same *AKE_label* value will be used for all control commands associated with a specific authentication procedure between a source and sink device.

number field specifies the step number of a specific control command to identify its position in the sequence of control commands making up an authentication procedure.

The initiator of an authentication procedure sets the value 1 for the initial AKE control command.
The value is incremented for each subsequent command that is part of the same authentication process.

When an AKE command must be fragmented for transmission (see the description of the *blocks_remaining* field below), each fragment will use the same value for the number field.
This field is optional and the devices that do not support this field shall set its value to 0000_2.

status field is used to notify the device issuing the command of the reason when the command results in a REJECTED response.
The device issuing the command sets the value of this field to 1111_2. If the responding device rejects the command, it overwrites the status field with a code indicating the reason for rejection.
If the command is accepted, the *status* field is set to 0000_2. For the interim response, the *status* field is set to 1111_2.

blocks_remaining field is used when a command is larger than the maximum command frame size that the target device can receive (A device issuing a command can determine the size of data field that the target device can accept using the AKE status command). When this occurs, the *data* field is fragmented into N blocks that are sent sequentially, each in one of N separate commands, where each command is small enough to be accommodated by the target device's command buffer. At a minimum, the buffer must be able to hold a command with at least a 32-byte data field. The size of the data field in the first N-1 fragments shall be the same size and a multiple of 16 bytes greater than or equal to 32 bytes.
Each of the N command frames is identical except for the values in the *blocks_remaining*, *data_length*, and *data* fields. For the first command, the *blocks_remaining* field is set to the value of N-1. In each successive command, the *blocks_remaining* field is decreased by one until it reaches zero, indicating the last command fragment.
Since the size of the command and response frames cannot exceed the 512-byte limit imposed by the underlying FCP transport, the case where a command must be fragmented can only occur when a target device has a command frame buffer capacity less than 512 bytes. Typically the command's size is within the target device's command frame buffer capacity and the command is sent without fragmentation and with a *blocks_remaining* value of zero.

Copyright © 1999, 1394 Trade Association. All rights reserved.

When an AKE_Info is transmitted using multiple Control Commands, a controller shall send each command only after receiving an ACCEPTED response for the previous command.

data_length field specifies the length of data field in bytes. Command and response shall have the same value in each *data_length* field.

- Responses to a command will use the same value for their respective *data_length* fields even when the response returns no data.
- If a response has some data when the response code is ACCEPTED, the corresponding command will have no data but the value of the *data_length* field shall be the same as that of response.

data field contains the data to be transferred.
The contents of the *data* field depend on the *AKE_ID* field and the *AKE_ID dependent field*.
For responses with a response code of REJECTED, there is no *data* field.

2.1.1.2 AKE status command frame

The format of AKE status command is as follows.

	msb							lsb
opcode	$0F_{16}$							
operand[0]	category = 0000_2 (AKE)				AKE_ID			
operand[1]	(msb)							
operand[2]	AKE_ID dependent field							
operand[3]								
operand[4]								(lsb)
operand[5]	FF_{16}							
operand[6]	F_{16}				status			
operand[7]	$7F_{16}$							(msb)
operand[8]	data_length							(lsb)

Figure 1.3 AKE status command frame

Both the Command and Response frames have the same structure.

The values of each field of the command and response frames are identical except for the *AKE_ID dependent*, *status*, and *data_length* field.
AKE_ID field specifies the format of *AKE_ID dependent field*.

Currently, only the encoding of *AKE_ID*=0 is defined. The other values, from 1_{16} to F_{16} are reserved for future definition.

In case of ***AKE_ID = 0***, ***AKE_ID dependent field*** contains such information as described below.

- Algorithms of AKE protocol which is supported by target.

status field is used by a device to query the state of another device. When the command is issued, the value of this field is set to 1111_2.
In the response, the target device overwrites this field with a value indicating its current situation.

Copyright © 1999, 1394 Trade Association. All rights reserved.

Value	status
0000_2	No error
0001_2 – 1110_2	Defined by DTLA
1111_2	No information

Table 1.2 definition of status field in AKE status command

data_length field specifies the target device's maximum *data* field capacity in bytes. When the status command is issued, the value of this field is set to $1FF_{16}$.
In the response, the target device overwrites this field with a value indicating its current situation. The minimum value to be supported will be defined by DTLA.

Copyright © 1999, 1394 Trade Association. All rights reserved.

2.1.2 *Vendor_dependent* SECURITY command frame

In case of *category* = E_{16}, the format of this command is as follows.

	msb							lsb
opcode	$0F_{16}$							
operand[0]	category = 1110_2				reserved_zero			
operand[1]	(msb)							
operand[2]	company_ID							
operand[3]								(lsb)
operand[4]	Vendor_dependent_data							
:								
operand[n]								

Figure 1.4 Vendor dependent command frame

Company_ID field shall contain the 24-bit unique ID obtained from the IEEE Registration Authority Committee (RAC).

The format and meaning of the *vendor_dependent_data* are specified by the company identified by *company_ID*.

The behavior of *vendor_dependent* commands is beyond the scope of this specification but it should be disclosed under some condition.

Copyright © 1999, 1394 Trade Association. All rights reserved.