Document 2009013

Audio and Music Data Transmission Protocol 2.3

April 24, 2012

Sponsored by:

1394 Trade Association

Accepted for Publication by:

1394 Trade Association Board of Directors

Abstract:

This specification defines Audio and Music data transmission over IEEE 1394 Bus. This specification includes definitions in the 1394 TA specification "Audio and Music Data Transmission Protocol Ver.1.0", "Enhancement to Audio and Music Data Transmission protocol Ver.1.0", "Audio and Music Data Transmission Protocol Ver.2.0" and IEC 61883-6. It also defines new extensions.

Keywords:
Audio and Music, IEC 61883-6, DVD-Audio, SACD, MIDI, Blu-ray Disc

1394 Trade Association Specification

Published by

1394 Trade Association
315 Lincoln, Suite E
Mukilteo, WA 98275

Printed in the United States of America

Table of Contents

List of figures

List of tables

Foreword (This foreword is not part of 1394 Trade Association Specification 2009013)

This specification defines Audio and Music data transmission over IEEE 1394 Bus. This specification includes definitions in the 1394 TA specification "Audio and Music Data Transmission Protocol Ver.1.0", "Enhancement to Audio and Music Data Transmission protocol Ver.1.0", "Audio and Music Data Transmission Protocol Ver.2.0" and IEC 61883-6. It also defines new extensions.

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

Max Bassler, Chair
Morten Lave, Vice-Chair
Dave Thompson, Secretary

Organization Represented	*Name of Representative*
Littelfuse	Max Bassler
TC Applied Technologies..	Morten Lave
Dap Technology	Richard Mourn
Texas Instruments..	Toni Ray
LSI	Dave Thompson
IPRA	Richard Davies

The A/V and PRO AUDIO Working Group, which revised and reviewed this specification, had the following members:

Morten Lave, Chair

Seiichi Hasebe, Pioneer

1. Overview

1.1 Purpose

This specification includes definitions in the 1394 TA specification "Audio and Music Data Transmission Protocol Ver.1.0", "Enhancement to Audio and Music Data Transmission Protocol Ver.1.0", "Audio and Music Data Transmission Protocol Ver.2.0" and IEC 61883-6. It also defines new extensions.

NOTE: With permission of IEC, this specification includes excerpts from IEC 61883-6, which is the improved version of 1394 TA specification "Audio and Music Data Transmission Protocol Ver.1.0". The original text in IEC 61883-6 may be subject to change because the document is still in the international standardization process.

1.2 Scope

This standard describes a protocol for the transmission of audio and music data over IEEE Std 1394-1995 or later. This includes the transport of IEC 60958 digital format, raw audio samples, and MIDI data.

This protocol can be applied to all modules or devices, which have any kind of audio and/or music data processing, generation and conversion function blocks. This specification deals only with the transmission of audio and music data; the control, status and machine readable description of these modules or devices should be defined outside of this specification according to each application area.

2. References

The following standards contain provisions, which through reference in this document, constitute provisions of this standard. All the standards listed are normative references. Informative references are given in Annex A. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below.

[R1] IEEE Std 1394-1995, Standard for a High Performance Serial Bus

[R2] IEEE Std 1394a-2000, Standard for a High Performance Serial Bus – Amendment 1

[R3] IEEE Std 1394b 2002, Standard for a High Performance Serial Bus – Amendment 2

[R4] IEEE Std 1394c 2006, Standard for a High Performance Serial Bus – Amendment 3

[R5] IEEE Std 1394d 2008, Standard for a High Performance Serial Bus – Amendment 4

[R6] IEC 61883-1, Consumer audio/video equipment – Digital interface – Part 1: General.

[R7] IEC 61883-6, Consumer audio/video equipment – Digital interface – Part 6: Audio and music data transmission protocol

[R8] IEC 60958-1, Digital audio interface – Part 1: General

[R9] IEC 60958-3, Digital audio interface – Part 3: Consumer applications

[R10] IEC 60958-4, Digital audio interface – Part 4: Professional applications

[R11] IEC 61937, Digital audio – Interface for non-linear PCM encoded audio bitstreams applying IEC 60958

[R12] IEC 62574, Audio, video and multimedia systems – General channel assignment of multichannel audio

[R13] IEEE Std 754-1985, Binary Floating-Point Arithmetic

[R14] MMA/AMEI RP-027, Specification for MIDI Media Adaptation Layer for IEEE-1394

[R15] Complete MIDI 1.0 Detailed Specification

[R16] TA 1999024, SMPTE Time Code and Sample Count Transmission Protocol Version.1.0

[R17] TA 2001012, AV/C Digital Interface Command Set General Specification, Version 4.1

[R18] TA 1999015, AV/C Command Set for Rate Control of Isochronous Data Flow 1.0

[R19] TA 1999014, Enhancement to Audio and Music Data Transmission Protocol 1.0

[R20] ASID Specification Draft Version 1.0, IFPI, RIAA and RIAJ

[R21] TA 2001003, Audio and Music Data transmission Protocol 2.0

[R22] TA 2001024, Audio and Music Data transmission Protocol 2.1

[R23] AES3-2009: AES standard for digital audio engineering - Serial transmission format for two-channel linearly represented digital audio data

[R24] AES58-2008: AES standard for digital audio - Audio applications of networks - Application of IEC 61883-6 32-bit generic data

3. Definitions

3.1 Conformance levels

3.1.1 expected: A key word 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.

3.1.2 may: A key word that indicates flexibility of choice with *no implied preference*.

3.1.3 shall: A key word indicating a mandatory requirement. Designers are *required* to implement all such mandatory requirements.

3.1.4 should: A key word indicating flexibility of choice with a strongly preferred alternative. Equivalent to the phrase *is recommended*.

3.1.5 reserved fields: A set of bits within a data structure that are defined in this specification as reserved, and are not otherwise used. Implementations of this specification shall zero these fields. Future revisions of this specification, however, may define their usage.

3.1.6 reserved values: A set of values for a field that are defined in this specification as reserved, and are not otherwise used. Implementations of this specification shall not generate these values for the field. Future revisions of this specification, however, may define their usage.

NOTE —The IEEE is investigating whether the "may, shall, should" and possibly "expected" terms will be formally defined by IEEE. If and when this occurs, draft editors should obtain their conformance definitions from the latest IEEE style document.

3.2 Glossary of terms

3.2.1 AM824: A 32-bit data field that has an 8-bit label and 24-bit data field defined in Audio and Music Data Transmission Protocol Version 1.0.

3.2.2 Audio Channel Cluster: Group of logical audio channels that carry tightly related synchronous audio information. A stereo audio stream is a typical example of a two-channel audio channel cluster.

3.2.3 Audio data stream: Transport medium that can carry audio information.

3.2.4 Byte: Eight bits of data.

3.2.5 Compound Data Block: The name for the Data Block that consists of AM824 data in any combination.

3.2.6 Conformant Data: A type of AM824 data that carries information equivalent to that defined in external specification such as IEC 60958 or MIDI.

3.2.7 IEEE: The Institute of Electrical and Electronics Engineers, Inc.

3.2.8 Isochronous: A term that indicates the essential characteristic of a time-scale or signal, such that the time intervals between consecutive instances either have the same duration or duration's that are integral multiples of the shortest duration. In the context of Serial Bus, "isochronous" is taken to mean a bounded worst-case latency for the transmission of data; physical and logical constraints that introduce jitter preclude the exact definition of "isochronous".

3.2.9 MIDI: Musical Instrument Digital Interface - an industry standard for the interconnection of music processing devices (e.g. keyboards, signal processors) and computers together. MMA (MIDI Manufactures Association, http://www.midi.org) or AMEI (Association of Musical Electronics Industry, http://www.amei.or.jp/) are contact points for the standard.

3.2.10 Music data: Data generally used for controlling a tone generator. The data defined in the MIDI specification, which may be called MIDI data, is an example of music data.

.**3.2.11 Node:** An addressable device attached to Serial Bus with at least the minimum set of control registers defined by IEEE Std 1394-1995.

3.2.12 Quadlet: Four bytes of data.

3.2.13 Stream: A time-ordered set of digital data originating from one source and terminating at zero or more sinks. A stream is characterized by bounded bandwidth requirements and by synchronization points, or time stamps, within the stream data.

3.3 Acronyms and abbreviations

A/M Protocol	Audio and Music Data Transmission Protocol.
ASID	Audio Software Information Delivery (See http://www.riaj.or.jp/standard/asid_e.html)
AV/C	Audio Video Control
DVD	Digital Versatile Discs (See http://www.dvdforum.org/index.htm)
SACD	Super Audio CD (See http://www.licensing.philips.com/)

4. Reference model for data transmission

This clause describes a reference model for data transmission.

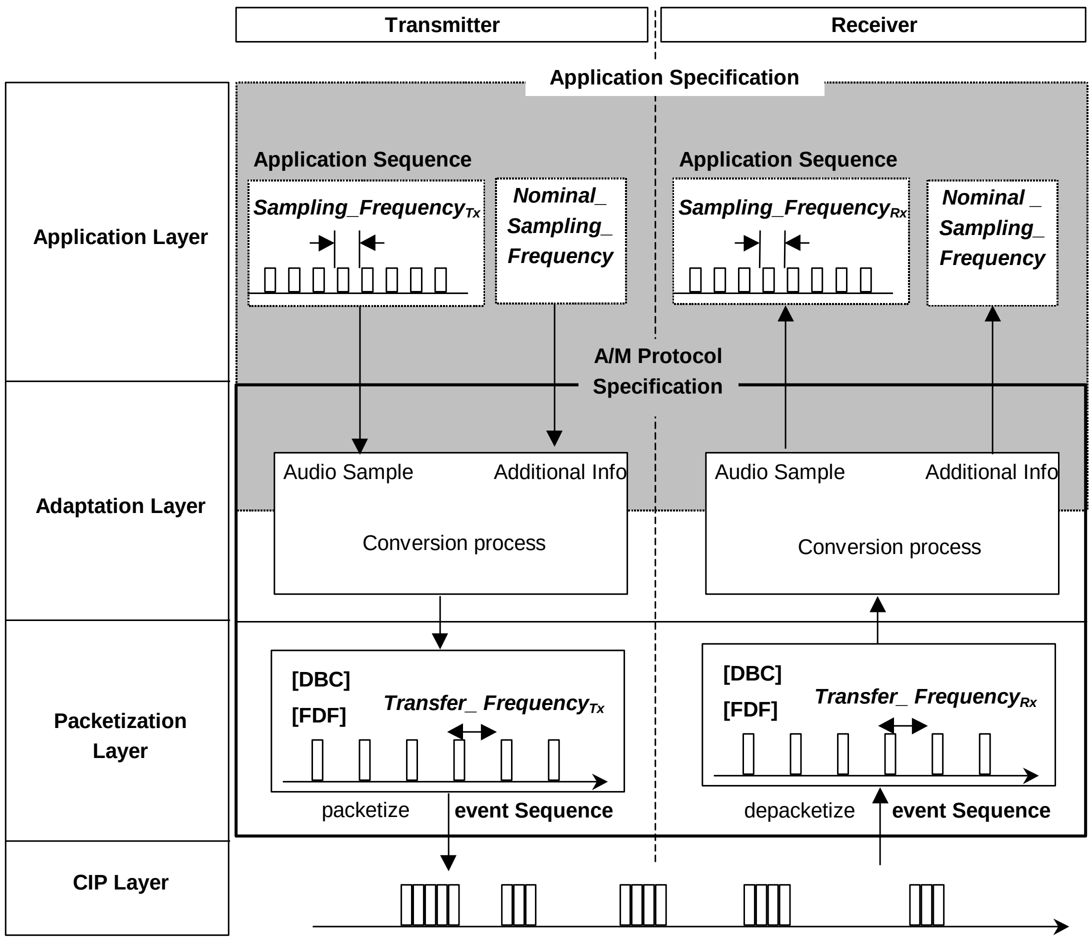

Figure 4-1 – Reference model for audio and music data transmission

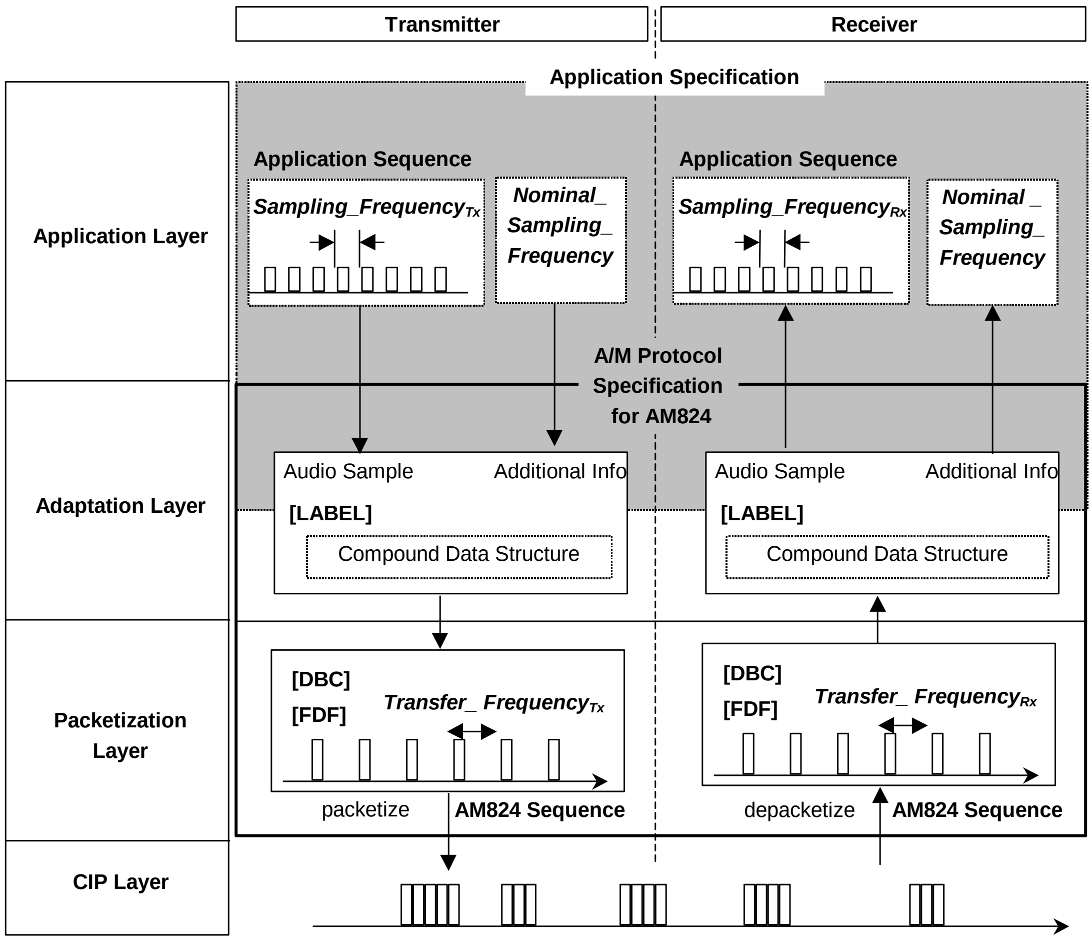

Figure 4-2 – Reference model for AM824 data transmission

Figure 4-1 – Reference model for audio and music data transmission gives an outline for audio data transmission from a transmitter to a receiver. It has four major layers denoted as CIP (Common Isochronous Packet) Layer, Packetization Layer, Adaptation Layer and Application Layer.

4.1 Application layer

Each application defines its own application sequence and the interface to the adaptation layer. The Application Sequence in Figure 4-1 – Reference model for audio and music data transmission is data in a format such as an audio signal format. The *Nominal_Sampling_Frequency* is the ideal sampling frequency for the Application Sequence. The range of *Sampling_Frequency* should be defined by the application. The audio signal at *Nominal_Sampling_Frequency* can be reproduced at the actual rate of *Sampling_Frequency* in operation. This means that the value of *Sampling_Frequency* may have some deviation and/or may vary in time in contrast with *Nominal_Sampling_Frequency*.

Additional Info in Figure 4-1 – Reference model for audio and music data transmission is any information other than events of a sequence (audio samples) being transmitted at a given rate.

4.2 Adaptation layer

Adaptation Layer defines a process to convert an Application Sequence to an Event Sequence and vice versa. The conversion process may not be required if an Application sequence and an Event Sequence have same structure. If an Event Sequence consists of events of 24-bit payload, such as AM824 Data defined in 8.1 AM824 Data, and if the bit length of an audio sample of the Application Sequence is not 24 bits, some conversion between *Sampling_Frequency* and *Tranfer_Frequency* may be required (see clause 11. AM824 adaptation processes). The *Transfer_Frequency* represents the frequency of occurrence of a Data Block, which is equivalent to a Cluster Event. The *Transfer_Frequency* is used for describing conceptual transmission model.

The transfer rate of an Event Sequence is 24 * *Transfer_ Frequency* [bits/sec] in case of AM824.

Generally, the Adaptation Layer is designed such that both the Application Sequence at *Sampling_Frequency* and its *Nominal_Sampling_Frequency* are carried. In this specification, *Nominal_Sampling_Frequency*, which would usually be one of the Ancillary Data items, is carried by SFC(Sampling Frequency Code) which is defined in clause 10. FDF definition for AM824 Data. The information in *Nominal_Sampling_Frequency* is necessary for using command based rate control or making a copy. On the other hand, *Sampling_Frequency* is necessary for clock based rate control. Although *Sampling_Frequency* is not explicitly transmitted, it can be estimated from SYT_INTERVAL and time stamps by the algorithm specified for the AM824 Data type.

An application specification defines the process (shown in the gray shaded area Figure 4-1 – Reference model for audio and music data transmission) to convert the application's signal (Application Sequence) to an Event Sequence. This document assumes that the application specification is an external document using the definition of an Event Sequence for the adaptation process. For several generic data types this document also defines the Adaptation Layer.

The adaptation to an Event Sequence is the point at which the packetization process interfaces to the application. The packetization process can be described as IEEE 1394 adaptation from the point of view that the data stream utilizes IEEE 1394 as its transport.

More details of this layer are described in clause 12. AM824 sequence adaptation layers.

4.3 Packetization layer

The AM824 Sequence is directly packetized to CIP or depacketized from CIP in the Packetization Layer.

The *Transfer_ Frequency* can be implicitly expressed by the output of a locked PLL circuit as shown in Figure 4-3 – Implementation example of receiver, instead of being explicitly denoted in the Packetization Layer.

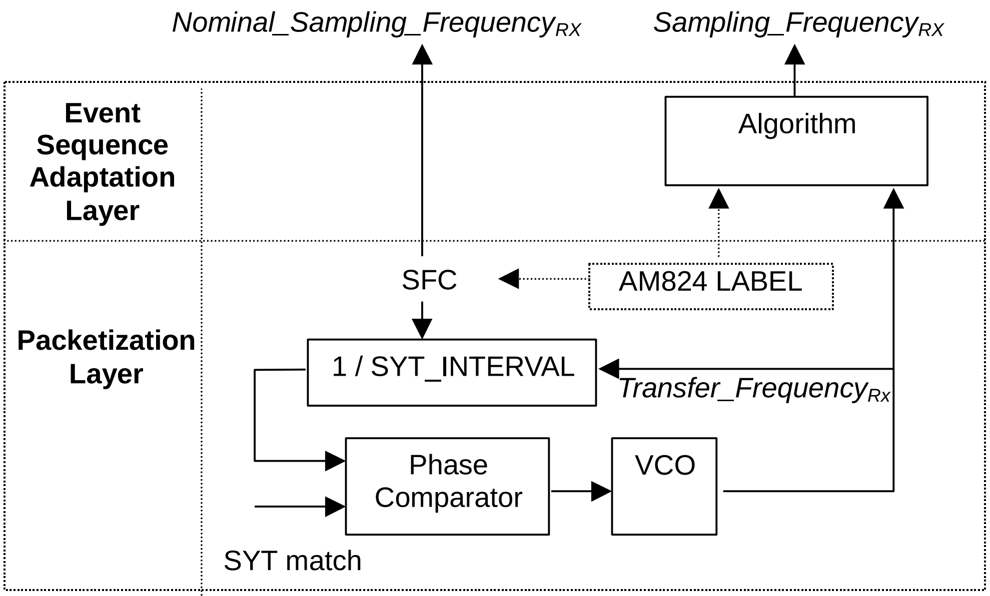

Figure 4-3 – Implementation example of receiver

5. Transport requirements

5.1 Arbitrated short Bus reset

All modules or devices which implement this A/M protocol should have the capability of "arbitrated short Bus reset" in order to prevent the interruption of audio and music data transmission when a Bus reset occurs.

5.2 Bit, byte, and quadlet ordering

This document adopts the ordering of bits, bytes, and quadlets for Bus packets according to the IEEE Std 1394-1995.

6. Packet header for audio and music data

This clause defines packet format in the CIP layer described in Figure 4-1 – Reference model for audio and music data transmission.

6.1 Isochronous packet header format

The header for an isochronous packet, which conforms to the A/M protocol shall have the same format given in Figure 6-1 – Isochronous packet header, which is part of the isochronous packet format defined in IEEE Std 1394-1995.

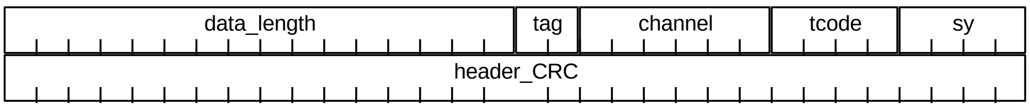

Figure 6-1 – Isochronous packet header

The isochronous packet header fields are defined with unique values that are specified in Table 6-1 – Isochronous packet header fields.

Table 6-1 – Isochronous packet header fields

Field	Value	Comments
Tag	01 b	This value indicates that a CIP header is included in the packet.
tcode	A_{16}	This value indicates that this is an isochronous data packet.
Sy	xx	This field is reserved. The transmitter shall set this field to 0_{16} unless specified by another application.

6.2 CIP header format

IEC 61883-1 defines a two-quadlet CIP header for a fixed length source packet with SYT field, repeated here for clarity as Figure 6-2 – Common isochronous packet (CIP) format. The CIP header format for an isochronous packet, which conforms to the Audio and Music Data transmission protocol, shall use this CIP header.

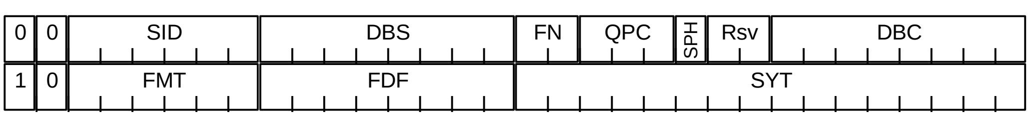

Figure 6-2 – Common isochronous packet (CIP) format

Table 6-2 – CIP fields define the fields with unique values that are specified by this protocol.

Table 6-2 – CIP fields

Field	Value	Comments
FMT	10_{16}	This value indicates that the format is for Audio and Music.
FN	0_{16}	
QPC	0_{16}	
SPH	0_{16}	
SYT	Xx	This field shall contain the time when the specified event is to be presented at a receiver.
FDF	Xx	This field is defined in 9. FDF Definition.

7. Packetization

7.1 Packet transmission method

When a non-empty CIP is ready to be transmitted, the transmitter shall transmit it within the most recent isochronous cycle initiated by a cycle start packet. The behavior of packet transmission depends on the definition of the condition in which “a non-empty CIP is ready to be transmitted.” There are two situations in which this condition is defined:

1. In order to minimize TRANSFER_DELAY, the condition of a non-empty CIP being ready for transmission is defined to be true if one or more data blocks have arrived within an isochronous cycle. This transmission method is called Non-Blocking transmission, and is described in 7.4.1 Non-Blocking transmission method

2. The condition of “non-empty CIP ready” can also be defined as true when a fixed number of data blocks have arrived. This transmission method is called Blocking transmission, and is described in 7.4.2 Blocking transmission method .

7.2 Transmission of timing information

A CIP without a source packet header (SPH) has only one time stamp in the SYT field. If a CIP contains multiple data blocks, it is necessary to specify which data block of the CIP corresponds to the time stamp.

The transmitter prepares the time stamp for the data block, which meets this condition:

mod(data block count, SYT_INTERVAL) = *0*   *(1)*

where

data block count is running count of transmitted data blocks;

SYT_INTERVAL denotes the number of data blocks between two successive valid SYTs, which includes one of the data blocks with a valid SYT. For example, if there are three data blocks between two valid SYT’s, then the SYT_INTERVAL would be 4.

The receiver can derive the index value from the DBC field of a CIP with a valid SYT using the following formula:

index = ***mod***((SYT_INTERVAL - ***mod***(DBC, SYT_INTERVAL)), SYT_INTERVAL)   *(2)*

where

index   is the sequence number ;

SYT_INTERVAL denotes the number of data blocks between two successive valid SYTs, which includes one of the data blocks with a valid SYT;

DBC   is the data block count field of a CIP.

The receiver is responsible for estimating the timing of data blocks between valid time stamps. The method of timing estimation is implementation-dependent.

7.3 Time stamp processing

A data block contains all data arriving at the transmitter within an audio sample period. The data block contains all data which makes up an "event".

The transmitter must specify the presentation time of the event at the receiver. A receiver for professional use must have the capability of presenting events at the time specified by the transmitter. A consumer-use or cost-sensitive receiver is not required to support this presentation-time adjustment capability.

If a function block receives a CIP, processes it and subsequently re-transmits it, then the SYT of the outgoing CIP shall be the sum of the incoming SYT and the processing delay.

The transmitter shall add TRANSFER_DELAY to the quantized timing of an event to construct the SYT. The TRANSER_DELAY value is initialized with the DEFAULT_TRANSFER_DELAY value. For professional use, TRANSFER_DELAY may be changed to achieve a shorter TRANSFER_DELAY value, according to the Bus configuration. Products for consumer use are not required to support the modification of TRANSFER_DELAY.

The DEFAULT_TRANSFER_DELAY value is 354.17 + 125 µs, which accommodates the maximum latency time of CIP transmission through an arbitrated short Bus reset.

7.4 Transmission control

7.4.1 Non-Blocking transmission method

Figure 7-1 – Non-blocking transmission method illustrates the Non-Blocking transmission method.

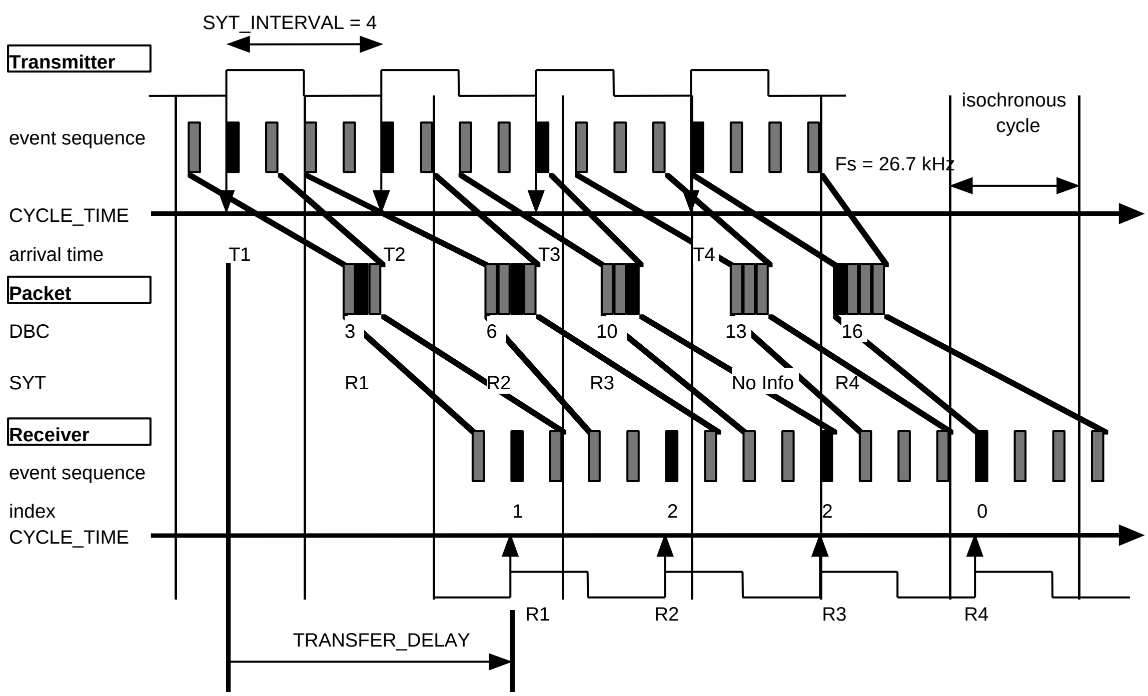

Figure 7-1 – Non-blocking transmission method

The transmitter shall construct a packet in every nominal isochronous cycle. Each packet shall comply with the following constraint:

0 <= N <= SYT_INTERVAL *(3)*

where

N is the number of events in the packet;

In normal operation the transmitter shall not transmit events late, and shall not transmit packets early. The resulting conditions may be expressed as follows:

Packet_arrival_time_L <= Event_arrival_time[0] + TRANSFER_DELAY *(4)*

Event_arrival_time[N-1] <= Packet_arrival_time_F *(5)*

where

Packet_arrival_time_F is the time (measured in µs) when the first bit of the packet arrives at the receiver;

Packet_arrival_time_L is the time (measured in µs) when the last bit of the packet arrives at the receiver;

Event_arrival_time[M] is the time (measured in µs) of the arrival at the transmitter of event M of the packet. The first event of the packet has M=0;

Figure 7-2 – Transmission parameters illustrates the transmission control rules as described in this section:

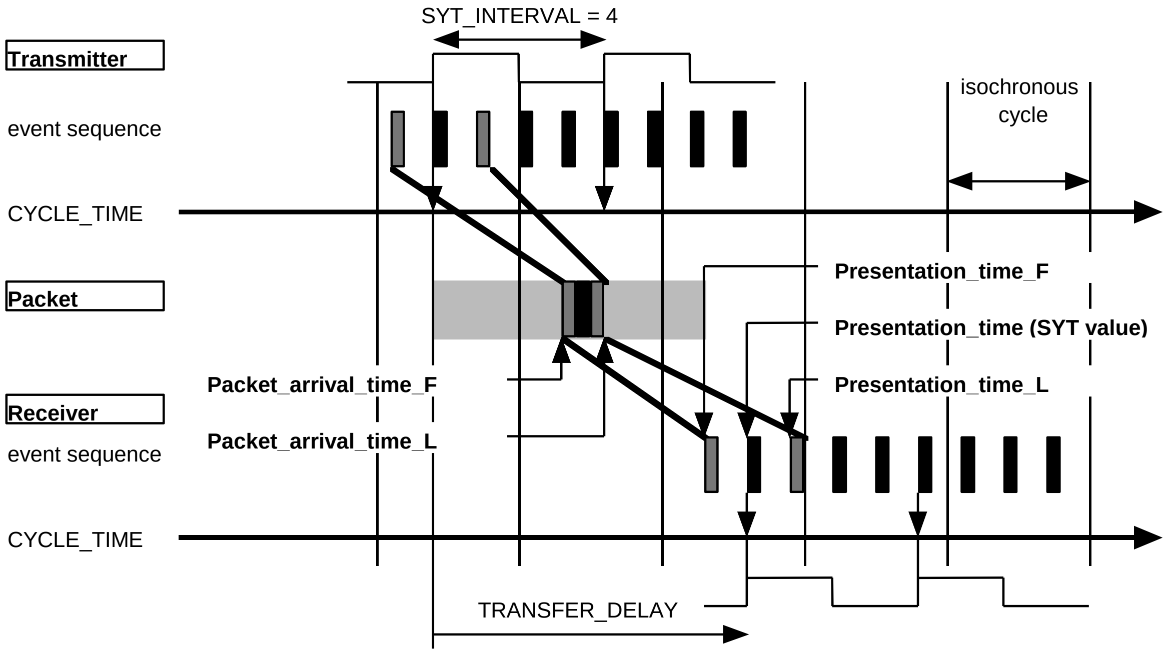

Figure 7-2 – Transmission parameters

In the event of lost opportunities to transmit non-blocking packets, a method of catching up may be provided. Refer to Annex C: Catching up in Non-Blocking Transmission method (informative) .

7.4.2 Blocking transmission method

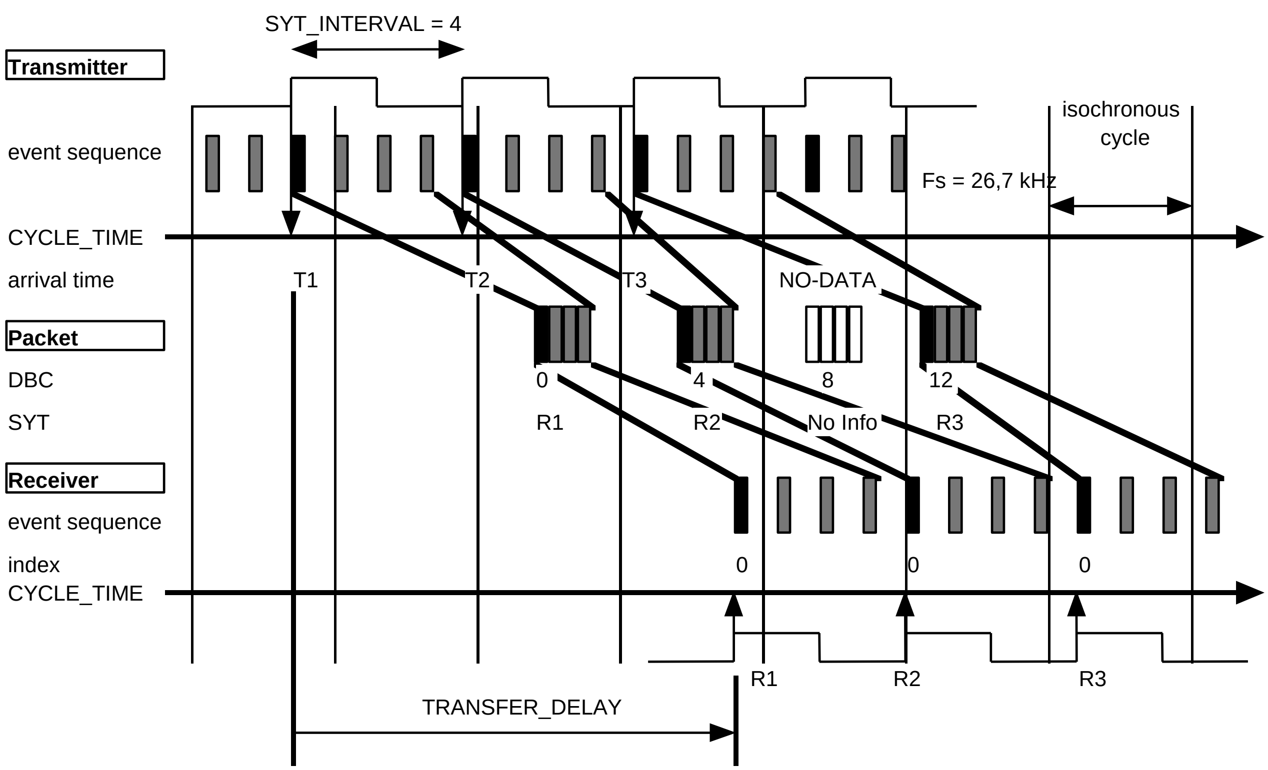

Figure 7-3 – Blocking transmission method

The blocking method may be used by a transmitter, which has only the ability to transmit packets of the same size. In order to indicate "no data", the transmitter may transmit an empty packet or a special non-empty packet which has the "NO-DATA" code in its FDF and has the same size of dummy data as a non-empty packet. The transmitter must set the time stamp of the first data block in a packet.

For blocking, the duration of the successive events in a CIP must be added to DEFAULT_TRANSFER_DELAY.

If a CIP contains N audio samples of a stream at Sampling Transmission Frequency (STF), then:

TRANSFER_DELAY >= DEFAULT_TRANSFER_DELAY + 1/STF * N * 1000

where

TRANSFER_DELAY	is the latency of transmission;
DEFAULT_TRANSFER_DELAY	is the initialized value of TRANSFER_DELAY;

STF is the sampling transmission frequency;

N is the number of audio samples in a CIP

The TRANSFER_DELAY for each STF when DEFAULT_TRANSFER_DELAY = 479.17µsec (= 354.17 + 125 µsec) is defined in Table 9-6 – TRANSFER_DELAY for blocking transmission.

It is recommended that the receiver have 250µs of extra buffer.

8. Event types

All the subformats described in this document shall use only 32-bit aligned events.

If multiple event sequences are synchronized, it is possible to convert the sequences into a single event, which consists of an ordered collection of those sequences, which occurrs at the same time. The ordered collection is called a cluster. A cluster consists of ordered units. In the case of data, a unit consists of a single sequence. In the case of a pack, the unit may consist of several sequences packed together. The number of units in a single cluster is called the dimension, and is denoted by CLUSTER_DIMENSION. Figure 8-1 – An example of cluster event illustrates these concepts.

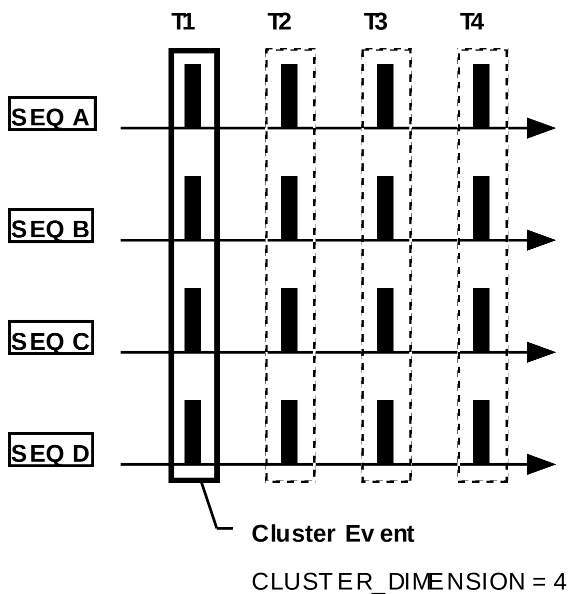

Figure 8-1 – An example of cluster event

In order to efficiently cluster non-32-bit aligned sequences which occur at the same time, the pack event type is defined. For example, four events of 24-bit data can be collected into a pack of three quadlets.

An event which is neither a cluster nor a pack is simply called data.

Only the pack and data types can be combined into units to make a cluster. All events in a cluster shall be of the same type.

UNIT_SIZE is the number of quadlets in a unit.

UNIT_DIMENSION is the number of sequences in a unit.

The UNIT_DIMENSION of data is always 1.

Figure 8-2 – An example of pack event cluster illustrates pack and cluster events.

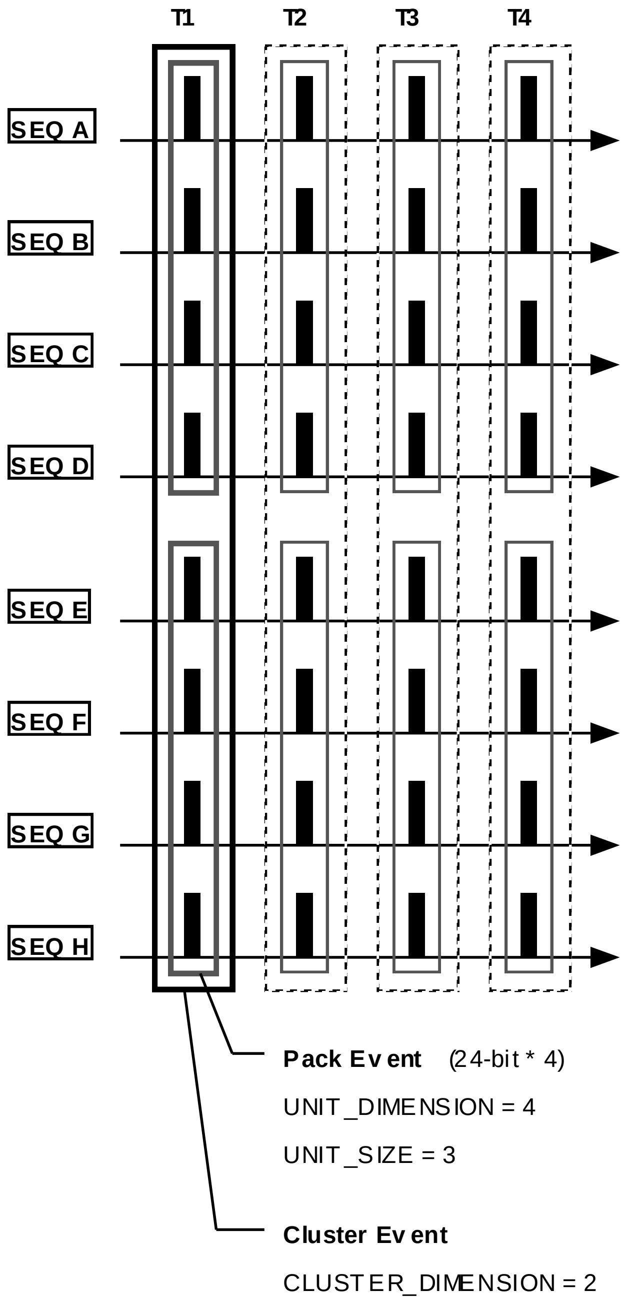

Figure 8-2 – An example of pack event cluster

Figure 8-3 – Pack event with 24-bit event sequence illustrates the structure of a pack which consists of four 24-bit event sequences (UNIT_DIMENSION = 4, UNIT_SIZE = 3).

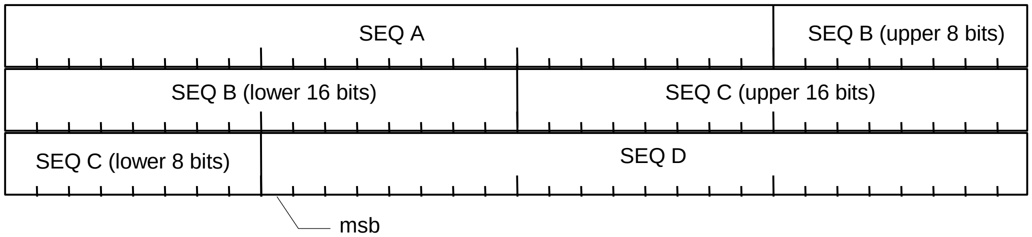

Figure 8-3 – Pack event with 24-bit event sequence

Since the cluster is an abstract event, only pack or data must be specified as an event type for a subformat. However, the DBS must reflect the size in quadlets of all cluster events in a data block. In case of a clustered sequence:

$$DBS = \sum_{n=0}^{(clusters-1)} (Unit\_Size_n * CLUSTER\_DIMENSION_n) \quad (6)$$

Where

CLUSTERS is the number of clusters in the event.

$Unit\_Size_n$ is the number of quadlets per unit of the n^{th} cluster.

$CLUSTER\_DIMENSION_n$ is the number of units per cluster of the n^{th} cluster.

Generally, the number of elementary sequences in a CIP is given by the following:

number of sequences = DBS * UNIT_DIMENSION / UNIT_SIZE *(7)*

For the pack illustrated in Figures 6 and 7, DBS = 6, CLUSTER_DIMENSION = 2, UNIT_DIMENSION = 4, UNIT_SIZE = 3.

The number of successive events in a CIP is equal to the number of Data Blocks in a CIP and given by:

NEVENTS_SUCCESSIVE = (data_length / 4 - CIPH_SIZE) / DBS *(8)*

where

data_length size of the payload of an isochronous packet (in bytes);

CIPH_SIZE size of the CIP header (in quadlets)

The ordering of sequences in an event is application-specific and is not within the scope of this specification. For example, the identification of audio channels in a multichannel transmission will be defined elsewhere.

8.1 AM824 Data

A 32-bit data consisting of an 8-bit label and 24-bit data is called AM824 data.

8.1.1 Generic Format

UNIT_SIZE = 1 quadlet/unit

UNIT_DIMENSION = 1 sequence/unit

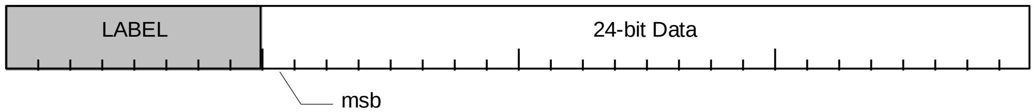

Figure 8-4 – Generic AM824 Data

A receiver capable of processing AM824 data must check the label for each AM824 data in a sequence being received.

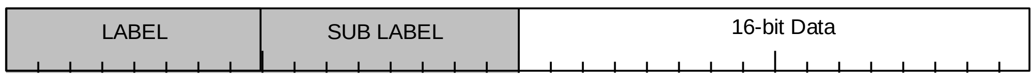

Figure 8-5 – AM824 Data with SUB LABEL

If an application requires many data types, SUB LABEL may be used to extend the number of data types defined by LABEL.

Table 8-1 – LABEL definition

Value	Description
00_{16} - $3F_{16}$	IEC 60958 Conformant
40_{16} - $4F_{16}$	Multi-bit Linear Audio
50_{16} - 57_{16}	One Bit Audio (Plain)
58_{16} - $5F_{16}$	One Bit Audio (Encoded)
60_{16} - 67_{16}	High Precision Multi-bit Linear Audio
70_{16} - $7F_{16}$	- reserved -
80_{16} - 83_{16}	MIDI Conformant
84_{16} - 87_{16}	- reserved -
88_{16} - $8B_{16}$	SMPTE Time Code Conformant
$8C_{16}$ - $8F_{16}$	Sample Count
90_{16} - BF_{16}	- reserved -
$C0_{16}$ – EF_{16}	Ancillary Data
$F0_{16}$ - FF_{16}	- reserved -

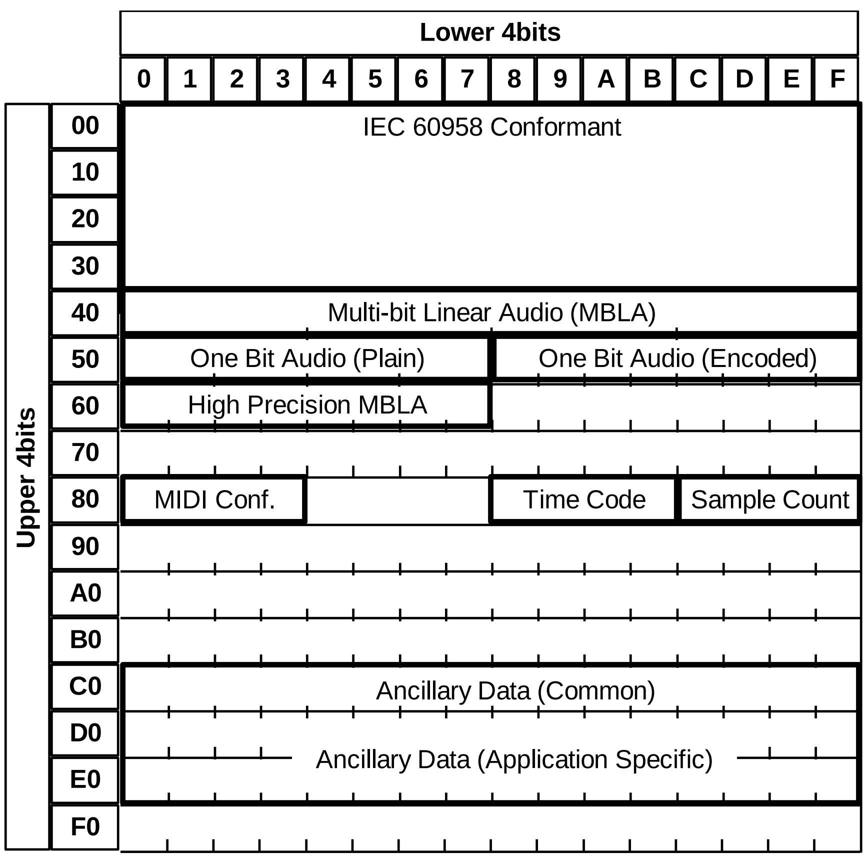

Figure 8-6 – AM824 LABEL allocation map (informative)

8.1.2 IEC 60958 Conformant Data

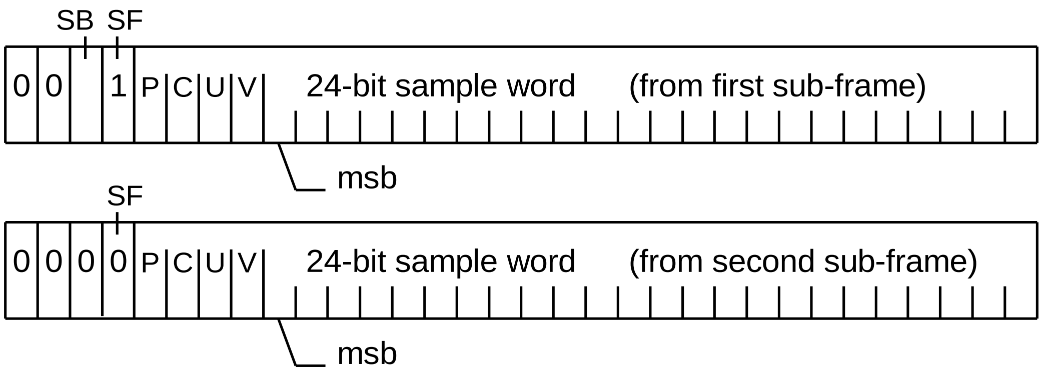

Figure 8-7 – IEC 60958 Conformant Data

Table 8-2 – PAC (Preamble code) definition

SB (Start of Block) and SF (Start of Frame) definitions				
LABEL	**SB**	**SF**	**Description**	**Equivalent IEC 60958 preamble codes**
00_{16} - $0F_{16}$	0	0	Second subframe of IEC 60958 frames 0 to 191	W,Y
10_{16} - $1F_{16}$	0	1	First subframe of IEC 60958 frames 1 to 191	M,X
20_{16} - $2F_{16}$	1	0	Reserved	-
30_{16} - $3F_{16}$	1	1	First subframe of IEC 60958 frame 0	B,Z

All information defined in the IEC 60958 standard is mapped into the data format shown in Figure 8-7 and Table 8-2. For each IEC 60958 frame, both sub-frames shall be transmitted together in the same event. The corresponding quadlets may be consecutive or non-consecutive. If multiple IEC 60958 streams are transmitted, then their sub-frames shall not be interleaved. Applications which use this data type shall follow the IEC 60958 standard.

8.1.3 Multi-bit Linear Audio (MBLA)

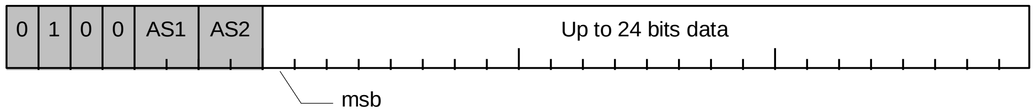

Figure 8-8 – MBLA data

The label field of MBLA has two fields for ASI (Application Specific Information). The definition of ASI2 depends on ASI1 value described in Table 8-3 – ASI1 definition.

Table 8-3 – ASI1 definition

Value	Description
00_2	Raw Audio. Sample word can be fed directly to a D/A converter. Ancillary Data may accompany. The definition of ASI2 is identical to VBL (Valid Bit Length) defined in A/M Protocol Ver.1.0.
01_2 - 11_2	Application Specific Information. Sample word may be fed directly to a D/A converter but in some processing required according to the application identified by application specific Ancillary Data which shall appear in the same Data Block. The definition of ASI2 field also shall be given by the application such as DVD-Audio described in 12.2 DVD-Audio and Blu-ray Disc described in 12.4 Blu-ray Disc.

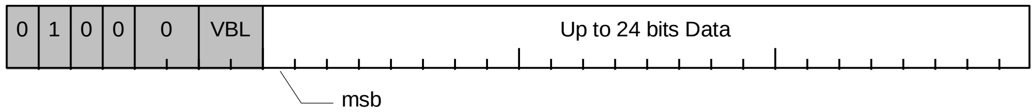

Figure 8-9 – Raw Audio Data

Table 8-4 – VBL (Valid Bit Length Code) definition

Value(Bin)	Description
00	24bits
01	20bits
10	16bits
11	- reserved -

The audio data shall be expressed in 24-bit 2's complement format. If the data active word length is less than 24 bits, the correct number of zero bits must be padded below the least significant bit to make a 24-bit data structure.

For example, a 20-bit audio data shall be placed in a 24-bit field as shown in Figure 8-10 – Bit alignment for 20 bits sample data (note the four zero pad bits at the right end of the structure):

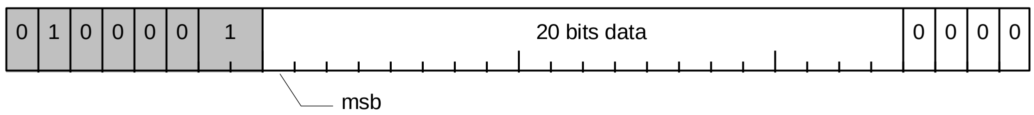

Figure 8-10 – Bit alignment for 20 bits sample data

Note: For audio data word lengths of less than 24 bits, the VBL indication can be used by receivers to determine if the data can be truncated to less than 24 bits without changing the value. If the word length is not known or variable, the data should be aligned at the most significant bit and the VBL code for 24 bits indication should be used.

8.1.4 One Bit Audio

One Bit Audio defines its own sampling frequency code (SFC).

Table 8-5 – LABEL definition for One Bit Audio (plain)

Value	Description
50_{16}	One Bit Audio Stream: Multi-Channel Cluster Start data
51_{16}	One Bit Audio Stream: Multi-Channel Cluster Continuation data
52_{16} - 57_{16}	- reserved -

Table 8-6 – LABEL definition for One Bit Audio (encoded)

Value	Description
58_{16}	DST: Encoded One Bit Audio stream
59_{16} - $5F_{16}$	- reserved -

8.1.5 MIDI Conformant Data

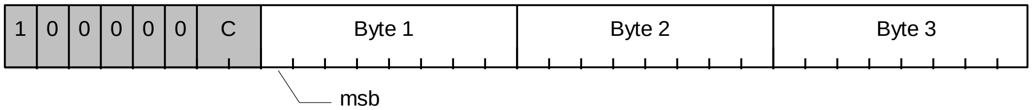

Figure 8-11 – MIDI Conformant Data

Table 8-7 – C (Counter) definition

Value (decimal)	Description
0	No Data (Byte 1 = Byte 2 = Byte 3 = 0)
1	Byte 1 is valid
2	Byte 1 and Byte 2 are valid
3	Byte 1, Byte 2 and Byte 3 are valid

If the CIP carries only MIDI Conformant data or cluster, and there is no MIDI data to be packed into a CIP, then the packet should be an empty packet rather than a packet of all "No Data" codes.

The "No Data" code defined in MIDI Conformant data may be used as "No Data" for other AM824 data types if necessary. The usage of "No Data" described above should be applied to the AM824 data types which use "No Data".

Figure 8-12 – "No Data" for MIDI Conformant Data illustrates the "No Data" structure.

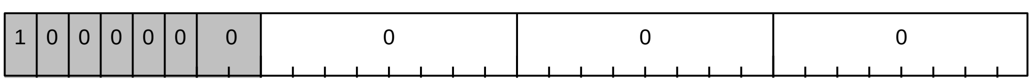

Figure 8-12 – "No Data" for MIDI Conformant Data

Successful implementation of MIDI Conformant Data may require additional information. Attention is drawn to MMA/AMEI Recommended Practice 027 [R14].

8.1.6 SMPTE Time Code Data

The SMPTE time code is defined in a separate document [R16].

8.1.7 Sample Count Data

The Sample count transmission is defined in a separate document [R16].

8.1.8 High Precision Multi-bit Linear Audio

The Multi-bit linear audio (MBLA) is limited to sample word up to 24 bits length. Linear PCM Audio data longer than 25 bit length and up to 196 bit length can be transmitted with High Precision Multi-bit linear audio.

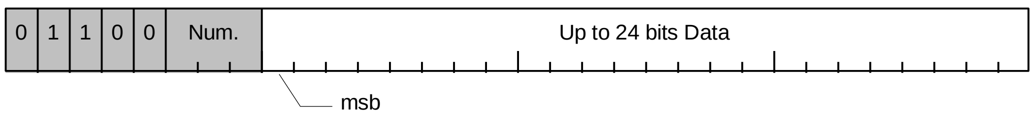

Figure 8-13 – High Precision Multi-bit linear audio data

The High Precision Multi-bit linear audio data use the LABEL from 60_{16} to 67_{16}. The label field of the High Precision Multi-bit linear audio has Num. (slot number) field. The definition of Num. field is described in Table 8-8 – Num. (slot number) definition.

Table 8-8 – Num. (slot number) definition

Value	Description
000_2	1st slot number (Num. = 0)
001_2	2nd slot number (Num. = 1)
010_2	3rd slot number (Num. = 2)
…	…
111_2	8th slot number (Num. = 7)

The High Precision Multi-bit linear audio data longer than 25 are divided into more than 2 quadlet sequence slots. The Num. (slot number) shall start with Num. = 0 (LABEL = 60_{16}) and be sequential. Figure 8-14 – Generic High Precision quadlet sequence shows generic quadlet sequence for the High Precision Multi-bit linear audio data.

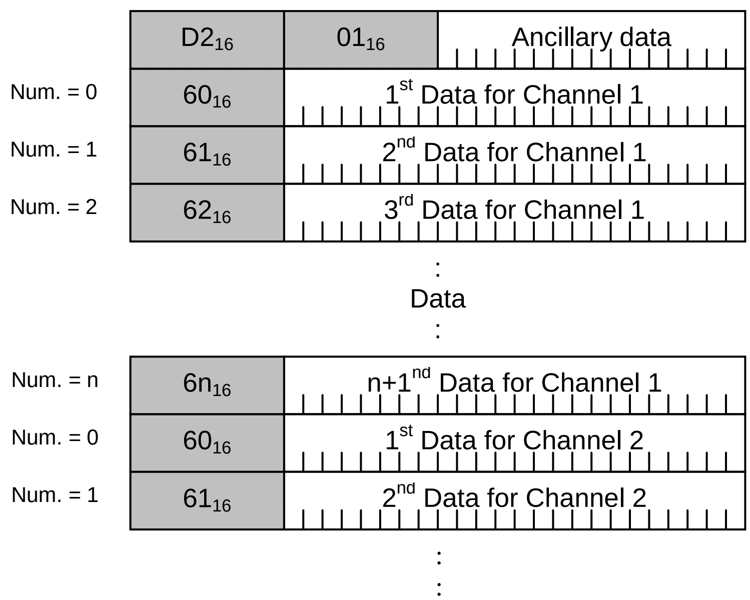

Figure 8-14 – Generic High Precision quadlet sequence

8.1.9 Ancillary Data

Generic Ancillary Data is illustrated in Figure 8-15 – Generic Ancillary Data. The definition of Byte 1, Byte 2, Byte 3 and transmission method, timing accuracy and interval for instance, should be given by each instance of Ancillary Data. It is recommended that all information carried by Ancillary Data be transmitted repeatedly in a reasonably short interval of time while the information is valid so that the receiver does not have to wait for the information. It is recommended that Byte 1 is defined as a SUB LABEL that specifies Byte 2 and Byte 3.

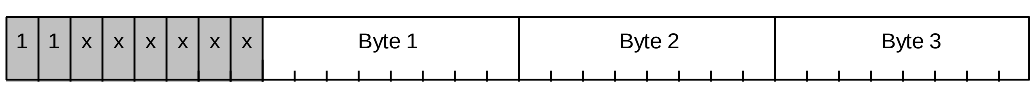

Figure 8-15 – Generic Ancillary Data

Table 8-9 – LABEL definition for Ancillary Data type

Value	Description
$C0_{16}$ - CF_{16}	Common Ancillary Data
$D0_{16}$ - EF_{16}	Application Specific Ancillary Data

8.1.9.1 Common Ancillary Data

Common Ancillary Data carries information common to all applications under a category of such as copyright information. The usage of this data is described in 11.3.1.2 Order rule.

Table 8-10 – LABEL definition for Common Ancillary Data

Value	Description
$C0_{16}$	ASID
$C1_{16}$ - CE_{16}	- reserved -
CF_{16}	Ancillary No-Data

8.1.9.1.1 Ancillary No-Data

Ancillary No-Data provides a No-Data event only for AM824 data that does not define its own No-Data. AM824 data types that define their own No-Data shall not use this Ancillary No-Data.

In order to determine whether the AM824 data type carries valid information, it is required that No-Data specify the AM824 data type to which it belongs. For this reason, the AM824 data type derived from a given No-Data should be identical to the AM824 data that carries valid information. The A/M Protocol Version 1.0 allows the use of No-Data defined in MIDI Conformant data.

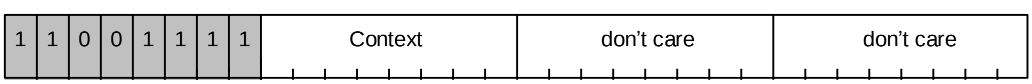

Figure 8-16 – Ancillary No-Data

Table 8-11 – CONTEXT definition

Value	Description
00_{16}	No-Data for IEC 60958 Conformant
01_{16} – $3F_{16}$	- reserved -
40_{16}	No-Data for Multi-bit Linear Audio
41_{16} – $4F_{16}$	- reserved -
50_{16}	No-Data for One Bit Audio (Plain)
51_{16} – 57_{16}	- reserved -
58_{16}	No-Data for One Bit Audio (Encoded)
59_{16} – $5F_{16}$	- reserved -
60_{16}	No-Data for High Precision Multi-bit Linear Audio
61_{16} – $7F_{16}$	- reserved -
80_{16} – 83_{16}	- reserved -
84_{16} –87_{16}	- reserved -
88_{16} –$8F_{16}$	- reserved -
$C0_{16}$ – CE_{16}	No-Data for each 7 different common Ancillary Data
CF_{16}	No-Data for unspecified type. This shall be used only for the purpose described in 11.3 Compound data block structure
$D0_{16}$ – EF_{16}	No-Data for each 32 different application specific Ancillary Data
$F0_{16}$ – FF_{16}	- reserved -

8.1.9.1.2 ASID(Audio Software Information Delivery)

ASID (Audio Software Information Delivery) defines transmission methods of ISRC, UPC/EAN and content usage (copyright assertion) information carried by the AM824 data.

The general format for ASID is shown in Figure 8-17 – General Format for ASID.

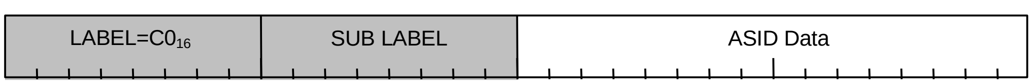

Figure 8-17 – General Format for ASID

The second byte SUB LABEL following the LABEL identifies the particular type of ASID data as shown in Table 8-12 – SUB LABEL definition for ASID.

Table 8-12 – SUB LABEL definition for ASID

SUB LABEL	Description
00_{16} – $0F_{16}$	UPC/EAN and ISRC
10_{16} – $1F_{16}$	Content Usage Information
20_{16} – FF_{16}	- reserved -

For details, see the reference document [R16] – ASID Specification.

8.1.10 Application Specific Ancillary Data

Application Specific Ancillary Data carries information specific to an application, which is transmitted along with the audio and music data. Examples are: mapping of sequence of a Compound Data Block to speaker location, microphone location or signal name.

Table 8-13 – LABEL definition for Application Specific Ancillary Data

Value	Description
$D0_{16}$	DVD-Audio
$D1_{16}$	SACD
$D2_{16}$	High Precision Multi–bit Linear Audio
$D3_{16}$	Blu-ray Disc
$D4_{16}$	Multi-bit Linear Audio (MBLA)
$D3_{16}$ – EF_{16}	- reserved -

The general format for the Application Specific Ancillary Data is shown in Figure 8-18 – General Format for Application Specific Ancillary Data:

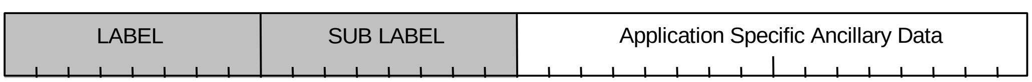

Figure 8-18 – General Format for Application Specific Ancillary Data

The first byte (“LABEL”) indicates that this data is for the Application Specific Ancillary Data of the type shown in Table 8-13 – LABEL definition for Application Specific Ancillary Data. The second byte (“SUB LABEL”) further identifies the particular data that follows. For details, see clause 12.2 for DVD-Audio, 12.3 for SACD, 12.4 Blu-ray and 12.5 Multi-bit Linear Audio (MBLA).

8.2 32-bit Floating Point Data

UNIT_SIZE = 1 quadlet/unit

UNIT_DIMENSION = 1 sequence/unit

Figure 8-19 – 32-bit floating point data illustrates the structure of 32-bit floating point data.

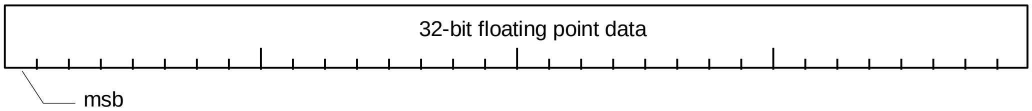

Figure 8-19 – 32-bit floating point data

8.3 24-bit * 4 Audio Pack

UNIT_SIZE = 3 quadlets/unit

UNIT_DIMENSION = 4 sequences/unit

Figure 8-20 – 24*4 Pack data illustrates the structure of a 24-bit * 4 audio pack:

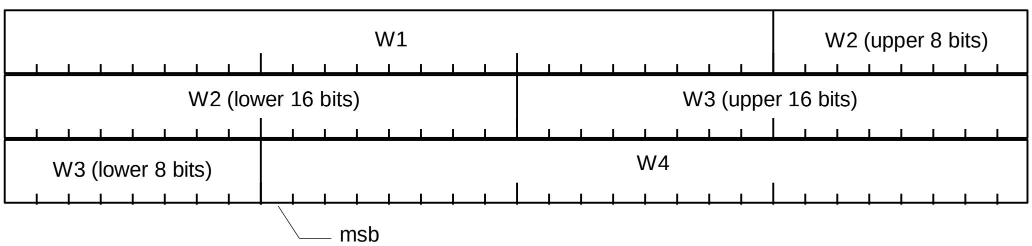

Figure 8-20 – 24*4 Pack data

The audio data must be expressed in 24-bit 2's complement. In the case of less than 24 bits, the correct number of zero bits must be padded below the least significant bit to make a 24-bit data structure. For an example of this, please refer to 8.1.3 Multi-bit Linear Audio (MBLA).

8.4 32-bit generic data

UNIT_SIZE = 1 quadlet/unit

UNIT_DIMENSION = 1 sequence/unit

Figure 8-21 illustrates the structure of 32-bit generic data.

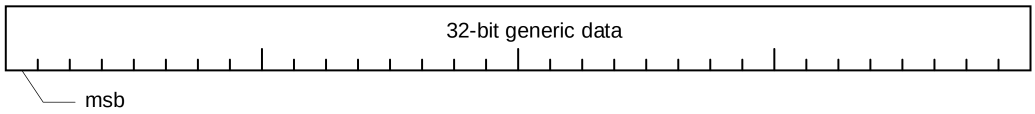

Figure 8-21 – 32-bit generic data

9. FDF Definition

Under the A/M packet format as described in 6.2 CIP header format, format dependent field (FDF) is used for specifying subformat type and additional information described in 4 Reference model for data transmission. Table 9-1 – Subformat and FDF allocations defines the subformat and Format dependent field (FDF) allocations.

Table 9-1 – Subformat and FDF allocations

Value	Description
0000 0xxx$_2$	Basic format for AM824
0000 1xxx$_2$	Basic format for AM824.Transmission rate may be controlled by a AV/C command set.
0001 0xxx$_2$	Basic format for 24-bit*4 Audio Pack
0001 1xxx$_2$	- reserved -
0010 0xxx$_2$	Basic format for 32-bit Floating-Point Data
0010 1xxx$_2$	- reserved -
0011 0xxx$_2$	Basic format for 32-bit generic data
0011 1xxx$_2$	- reserved -
0100 0xxx$_2$ - 1111 1110$_2$	- reserved -
1111 1111$_2$	Packet for NO-DATA

Each subformat may use a “cluster” for synchronized multiple sequences unless otherwise specified.

9.1 Special Format

1	1	1	1	1	1	1	1

Figure 9-1 – FDF code for NO-DATA packet

The transmitter must use the FDF code shown in Figure 9-1 – FDF code for NO-DATA packet when a packet is a NO-DATA packet only for blocking transmission. The transmitter must not use this FDF code for non-blocking transmission. The receiver must ignore all the data in a CIP with this FDF code.

9.2 Basic Format

Table 9-2 – DBS for AM824 and 32-bit floating point data

Value (decimal)	Description
0	CLUSTER_DIMENSION = 256
1 – 255	CLUSTER_DIMENSION = DBS

Table 9-3 – DBS for 24*4 audio pack

Value (decimal)	Description
3 – 255	CLUSTER_DIMENSION = DBS/3

Figure 9-2 – Generic FDF definition illustrates a generic FDF definition.

0	0	EVT	0	SFC

Figure 9-2 – Generic FDF definition

Table 9-4 – Event type (EVT) code definition

Value (decimal)	Description
0	AM824 Data
1	24-bit * 4 Audio Pack
2	32-bit Floating-Point Data
3	- reserved -

Table 9-5 – Default SFC table

Value	Description	
	SYT_INTERVAL	*Nominal_Sampling_Frequency*
00_{10}	8	32kHz
01_{10}	8	44.1kHz
02_{10}	8	48kHz
03_{10}	16	88.2kHz
04_{10}	16	96kHz
05_{10}	32	176.4kHz
06_{10}	32	192kHz
07_{10}	- reserved -	- reserved -

Table 9-6 – TRANSFER_DELAY for blocking transmission

Value	TRANSFER_DELAY
00_{10}	479.17 + 250.00 = 729.17 [µ sec]
01_{10}	479.17 + 181.41 = 660.58 [µ sec]
02_{10}	479.17 + 166.67 = 645.84 [µ sec]
03_{10}	479.17 + 181.41 = 660.58 [µ sec]
04_{10}	479.17 + 166.67 = 645.84 [µ sec]
05_{10}	479.17 + 181.41 = 660.58 [µ sec]
06_{10}	479.17 + 166.67 = 645.84 [µ sec]
07_{10}	- reserved -

If a packet of AM824 data contains only IEC 60958 conformant data and a transmitter functions as a gateway, then the transmitter should estimate the sample transmission frequency for the SFC rather than copying the sampling frequency code embedded in the original IEC 60958 data.

Equation (9) below can be used to determine the required Bus bandwidth allocation. The required isochronous bandwidth is given below:

$$BW = (\text{int}\,(\max(Fs)/8000) + 1) * \sum_{n=0}^{\text{clusters-1}} (UNIT\_SIZE_n * CLUSTER\_DIMENSION_n) * 8000 \quad (9)$$

where

BW is the required isochronous bandwidth (in quadlet/sec);

FS	is the sample rate (in Hz);
$UNIT\_SIZE_n$	is the number of quadlets in a unit of the nth cluster;
$CLUSTER\_DIMENSION_n$	is the number of units in the nth cluster.
CLUSTERS	is the number of clusters in an event.

10. FDF definition for AM824 Data

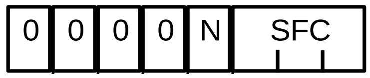

Figure 10-1 – Structure of FDF for AM824 data type

10.1 N-flag

The N-flag as shown in Figure 10-1 – Structure of FDF for AM824 data type shall be used to select the AM824 LABEL space and adaptation process described in clause 10.4 Command based rate control mode (FDF = $00001xxx_2$).

Any AM824 data type shall occupy the same space in both LABEL spaces. An application may use only one of two LABEL spaces by giving a fixed value to the N-flag. Only an AM824 data type that owns the LABLE space or Application Specific Ancillary Data, which is defined in clause 8.1.10 Application Specific Ancillary Data, can inhibit the use of one of the LABEL spaces.

10.2 Supplementary SFC definition

In A/M Protocol Version. 1.0, there is only one SFC table that specifies both *Nominal_Sampling_Frequency* and SYT_INTERVAL.

In this specification, the SFC definition is changed so that a new AM824 Data Type which is defined after the A/M Protocol Version. 1.0 may define its own SFC table. In order to keep compatibility with the A/M Protocol Version. 1.0, in the case of FDF = $0000\ 0xxx_2$, the default SFC table shall be identical to the table defined in A/M Protocol Version. 1.0. Only a new AM824 Data type may override the default SFC table.

The empty packet defined in [R6] shall use the default SFC table.

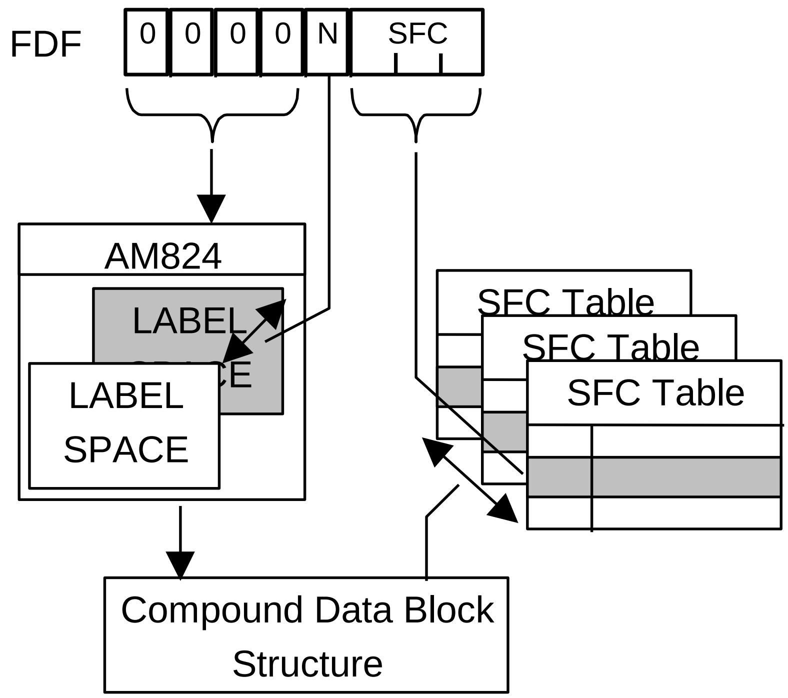

Figure 10-2 – SFC interpretation

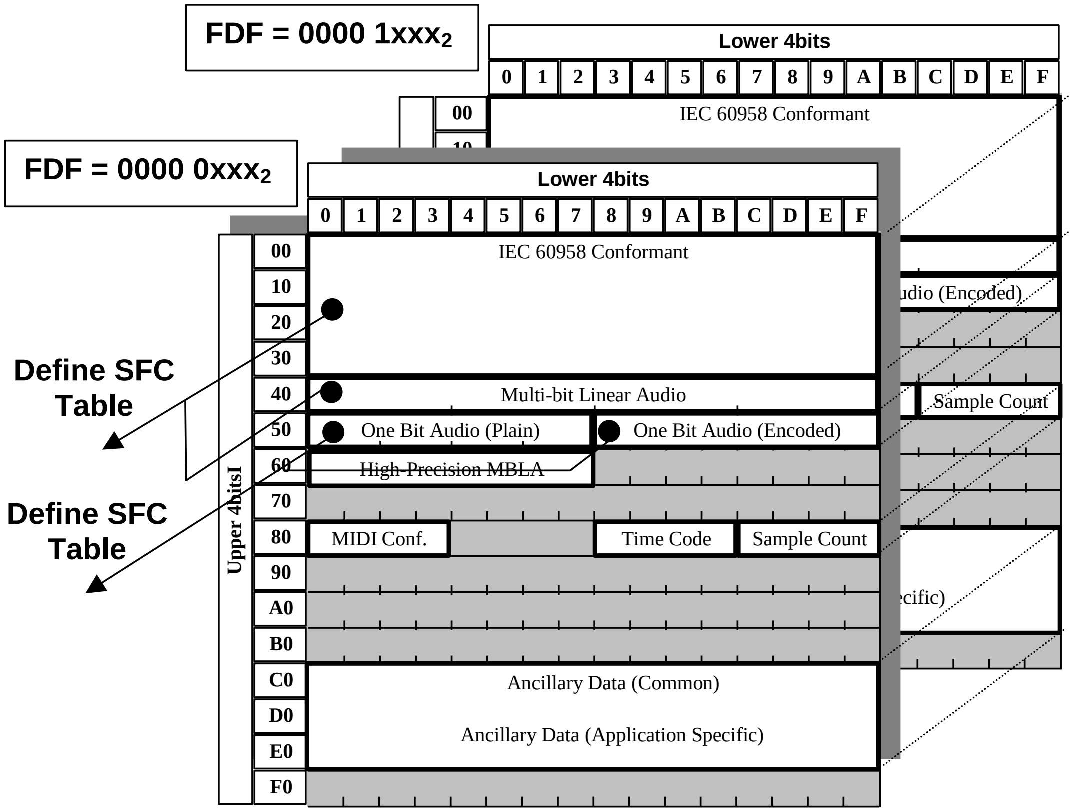

Figure 10-3 – FDF for AM824 and AM824 LABEL space (informative)

10.3 Clock based rate control mode (FDF = 0000 0xxx$_2$)

This FDF value, which is defined in A/M Protocol Version 1.0, is interpreted to indicate that the data transmission rate is controlled by a transmission clock reproduced by means of a timestamp.

The meaning of this FDF value is not changed.

10.3.1 Default SFC table for (FDF = 0000 0xxx$_2$)

Table 10-1 – Default SFC table for FDF = 0000 0xxx$_2$

Value	Description	
	SYT_INTERVAL	*Nominal_Sampling_Frequency*
00_{10}	8	32kHz
01_{10}	8	44.1kHz
02_{10}	8	48kHz
03_{10}	16	88.2kHz
04_{10}	16	96kHz
05_{10}	32	176.4kHz
06_{10}	32	192kHz
07_{10}	- reserved -	- reserved -

The TRANSFER_DELAY for Blocking Transmission, in the case of DEFAULT_TRANSFER_DELAY = 479.17µsec = (354.17 + 125)µsec, corresponds to the default SFC table as given in Table 10-2 – TRANSFER_DELAY for blocking transmission.

Table 10-2 – TRANSFER_DELAY for blocking transmission

Value	TRANSFER_DELAY
00_{10}	479.17 + 250.00 = 729.17 [µ sec]
01_{10}	479.17 + 181.41 = 660.58 [µ sec]
02_{10}	479.17 + 166.67 = 645.84 [µ sec]
03_{10}	479.17 + 181.41 = 660.58 [µ sec]
04_{10}	479.17 + 166.67 = 645.84 [µ sec]
05_{10}	479.17 + 181.41 = 660.58 [µ sec]
06_{10}	479.17 + 166.67 = 645.84 [µ sec]
07_{10}	- reserved -

10.4 Command based rate control mode (FDF = 00001xxx$_2$)

This new allocated FDF value indicates that the data transmission rate is controlled by a command set such as *AV/C Command Set for Rate Control of Isochronous Data Flow* [R18].

This transmission mode can be used for reproducing an application sequence at a receiver or for high-speed data transfer without using a timestamp in the SYT field.

If the timing information is available, the transmitter should provide the correct timestamp in the SYT field according to the integer multiplier *n* so that the clock based rate controlled receiver can receive the data transmitted in this mode.

$SYT\_INTERVAL_{N\text{-}flag=1} = SYT\_INTERVAL_{N\text{-}flag=0} * n$ ($n >= 1$)

where $SYT\_INTERVAL_{N\text{-}flag=1}$ and $SYT\_INTERVAL_{N\text{-}flag=0}$ denote SYT_INTERVAL specified by the SFC table in the cases in which FDF = 0000 $1xxx_2$ and FDF = 0000 $0xxx_2$ respectively. The integer multiplier *n* is obtained by a command.

10.4.1 Default SFC table for (FDF = 0000 $1xxx_2$)

Table 10-3 – Default SFC table for FDF = 0000 $1xxx_2$

Value (decimal)	*Nominal_Sampling_Frequency*	SYT_INTERVAL	*Sampling_Frequency*
0	32 kHz	8 * *n*	32 kHz * *n*
1	44.1 kHz	8 * *n*	44.1 kHz * *n*
2	48 kHz	8 * *n*	48 kHz * *n*
3	88.2 kHz	16 * *n*	88.2 kHz * *n*
4	96 kHz	16 * *n*	96 kHz * *n*
5	176.4 kHz	32 * *n*	176.4 kHz * *n*
6	192 kHz	32 * *n*	192 kHz * *n*
7	- reserved -	- reserved -	- reserved -

The DBS of an event is independent of the transfer speed.

11. AM824 adaptation processes

This clause describes typical methods of adaptation to an AM824 Sequence.

11.1 Basic sequence conversion

Transfer_Frequency is identical to *Sampling_Frequency* (transfer frequency of the application sequence such as audio) to be packetized if each event in the application sequence (each audio sample) is stored in one unit such as one AM824 Data of an AM824 Sequence.

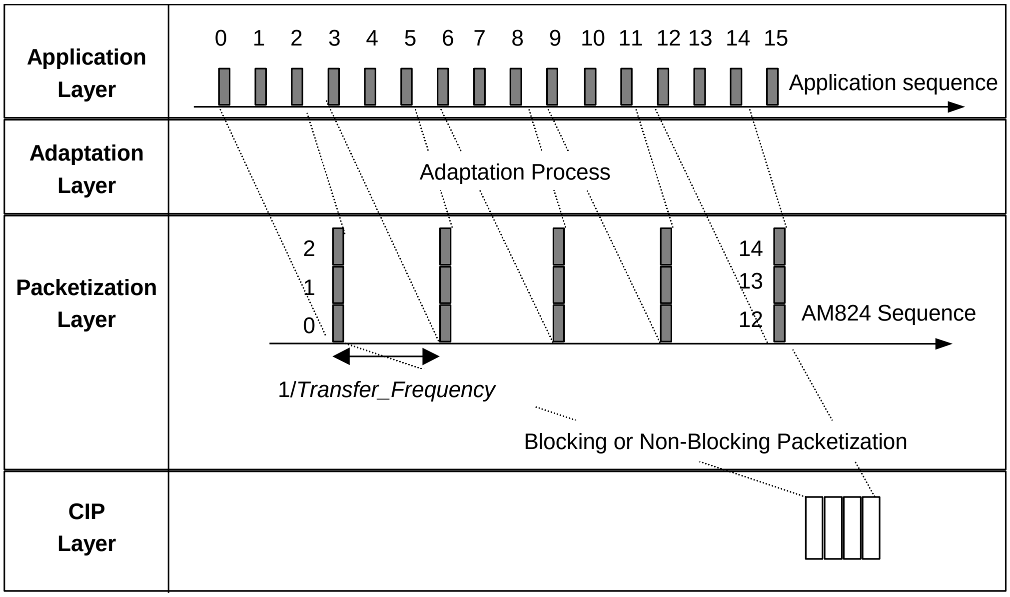

Figure 11-1 – Adaptation to AM824 sequence

Figure 11-1 – Adaptation to AM824 sequence describes an example of an adaptation process in which each event of the application sequence is 8 bits in length and three events are stored in a single AM824 data which has a 24-bit payload. In this case, the relation between *Sampling_Frequency* and *Transfer_Frequency* is expressed by

Sampling_Frequency = L * *Transfer_Frequency*

where L = 3.

The parameters *Sampling_Frequency*, *Transfer_Frequency* and L can not be specified independently. All of them are specified by the SFC code selected by the AM824 data type.

11.2 Sequence multiplexing

If the event occurrence rate of an application sequence is less than half of the rate of the Compound Data Block, one single Event sequence can carry more than one application sequence by multiplexing the application sequence into a single Event sequence assigned to the Compound Data Block. In this case each multiplexed application sequence is identified by its DBC (Data Block Count).

If the AM824 Sequence defines No-Data for padding, even an application sequence, which is asynchronous to *Transfer_Frequency* can be adapted to the AM824 sequence. One significant example of this case is the adaptation of a MIDI data stream (Application Sequence) to a MIDI Conformant sequence (AM824 Sequence).

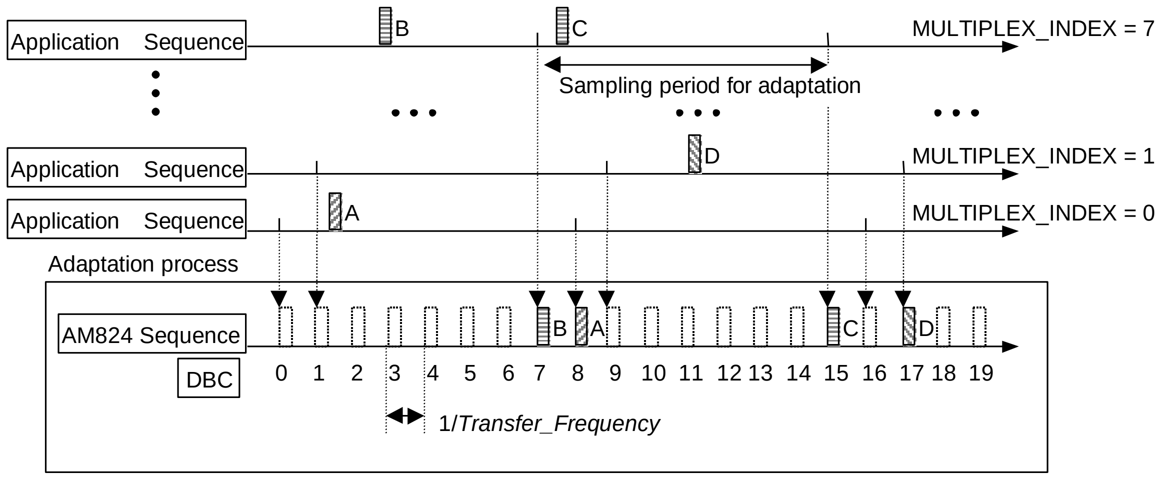

Figure 11-2 – Asynchronous sequence multiplexing

An application that uses this multiplexing shall define MULTIPLEX_NUMBER to be a power of 2.The MULTIPLEX_NUMBER is defined in conjunction with the LABEL definition because the place for carrying the MULTIPLEX_NUMBER information is not defined in this document. This definition will be overridden by a future specification if it defines a method of carrying the MULTIPLEX_NUMBER.

The identifier for a multiplexed sequence denoted by MULTIPLEX_INDEX is given by MULTIPLEX_INDEX = mod (DBC, MULTIPLEX_NUMBER).

11.3 Compound data block structure

Compound Data Block is the name for the Data Block that consists of AM824 data in any combination, if all the AM824 data in the Data Block specify the same SFC table. (Note that the SFC value in a CIP specifies the entry of the SFC table selected according to AM824 Data type that defines the SFC table.)

Thus the cluster, which is equivalent to a Data Block in the context of AM824 data, can be referred to as a Compound Cluster.

Each sequence carried by a Compound Data Block is uniquely identified by the location of events in the Compound Block.

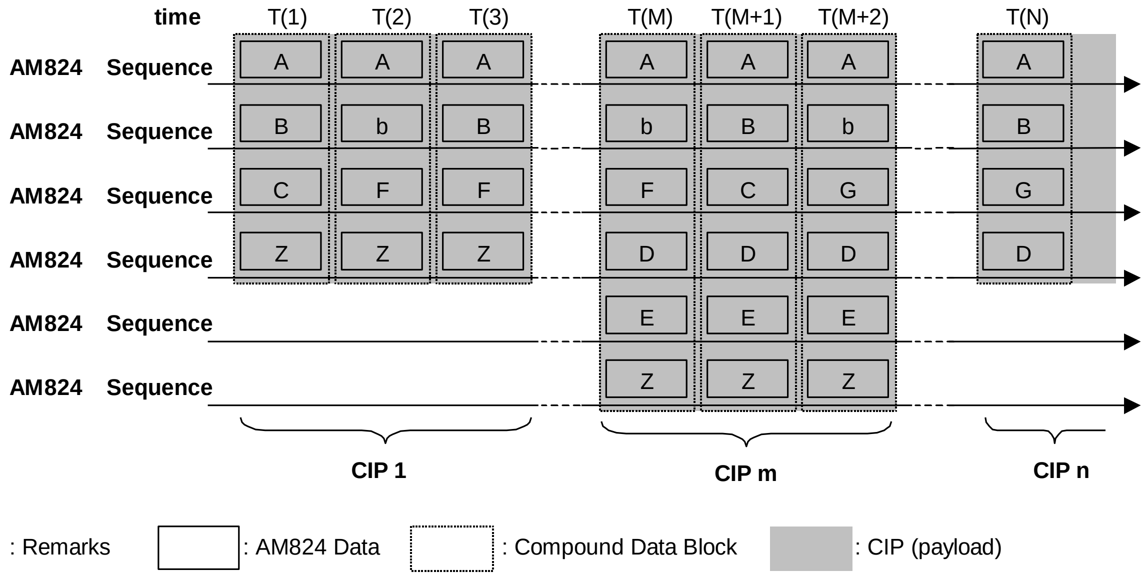

Figure 11-3 – Example of compound data block

An example of usage of Compound Data Block is illustrated in Figure 11-3 – Example of compound data block.

The capital letter, 'B' for example, in the box of AM824 Data, represents the box's data type. The small letter, 'b' for example, in the box of AM824 Data, denotes "No Data" for same data type.

DBS (Data Block Size) or CLUSTER_DIMENSION may vary in time. Also, the AM824 Data type described in the LABEL field of each event may vary in time.

11.3.1 Compound data structure rule

A/M Protocol Version 1.0 allows any order of AM824 data type in a Compound Data Block. In order to maintain minimum connectivity, this clause defines rules for the Compound Data structure, or in another words, a rule for AM824 sequence configuration. Also, some recommendations for implementation are described.

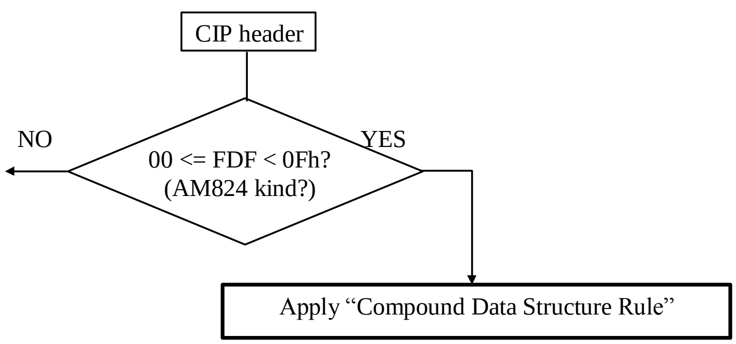

Figure 11-4 – Condition of AM824 rule

11.3.1.1 Size rule

The size of Compound Data should be an even number of quadlets.

If the number of quadlets in a sequence required by an application is not an even number, an unspecified sequence (sequence of Ancillary No-Data with CONTEXT = CF_{16}) should be added to make the number of quadlets in the sequence even. Figure 11-3 – Example of compound data block shows a Compound Data Block compliant to this rule where the event denoted by “Z” is interpreted as Ancillary No-Data with CONTEXT = CF_{16}. As long as the number of quadlets in a sequence is even, any number of unspecified sequences may be added.

11.3.1.2 Order rule

Application Specifier is either Application Specific Ancillary Data or any Common Ancillary Data except Ancillary No-Data for non-Ancillary Data. *Content Data* is any AM824 data other than *Application Specifier*.

A Compound Data Block starts with zero or only one *Unspecified Region* followed by zero or one or more *Specified Region*(s). *Unspecified Regi*on includes only *Content Data*. *Specified Region* starts with one or more *Application Specifiers* followed by one or more *Content Data* before encountering the next *Application Specifier* or the end of the Compound Data Block.

A sequence of *Application Specifiers* may contain both Common Ancillary Data and Application Specific Ancillary Data by multiplexing.

The order of the Content Data in an Unspecified Region shall be determined by following formula:

IEC 60958 Conformant Data < Multi-bit Linear Audio < MIDI Conformant Data < SMPTE Time Code < Sample Count.

Within an Unspecified Region the same data type should occupy a contiguous area.

The order inside a Specified Region is defined by the application specified in the Application Specific Ancillary Data. The Specified Region shall have none or only one Common Ancillary Data or one or more Application Specific Data for the same application.

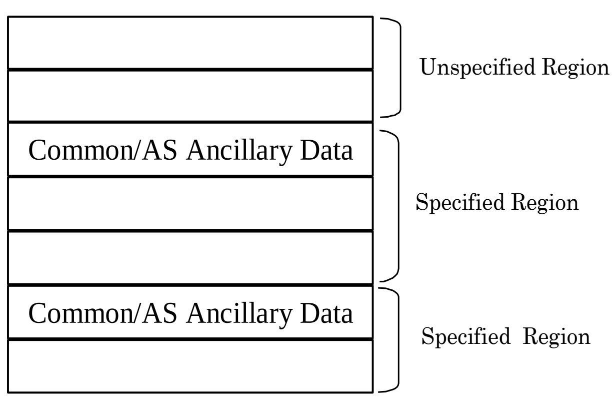

Figure 11-5 – Generic compound data block structure

IEC 60958 Conformant L-ch
IEC 60958 Conformant R-ch
MBL Audio Data 1-ch
MBL Audio Data 2-ch
MBL Audio Data 3-ch
MBL Audio Data 4-ch
MIDI Conformant Data

Figure 11-6 – Example of unspecified region structure

11.3.1.3 Recommendation: general

Because 2-channel stereo audio is widely accepted, it is highly recommended that for devices which transmit audio in any format, the first 2 sequences be linear audio either in IEC 60958 Conformant or Raw Audio. The first sequence should be left and second should be right. If a transmitter is a monaural audio device, it may send the audio in left channel and silent data in the right, or send the audio in both left and right. It is implementation-dependent.

If a transmitter is a multi-channel audio device, it may send downmixed in 2-channel stereo audio in addition to the multi-channel audio.

11.3.1.4 Recommendation for transmitter

A.1.1 DBS (Data block size in quadlets) should be greater or equal to 2. An even number is the most preferable.

A.1.2 At the top of the Data Block of mixed Audio and Music data, Stereo Left Channel, then Right Channel should be transmitted.

A.1.3 In Data Blocks of multichannel audio data, the first two quadlets should be the main channels corresponding to Stereo Left and Right Channel.

A.1.4 Recommendation for stream change method is as follows.

When the contents of stream are changed, it is preferable to insert Ancillary No-Data or Empty packets at the change point of the stream.

The change point of the stream is not a pause of each tune in the CD album, but it implies that at the point, some change of e.g. compression methods occurs.

The purpose of insertion of Ancillary No-Data or Empty packets is to prevent losing the end portion of the previous stream and the beginning of the next stream.

The general recommendation method is described as follows.

It is desirable to output Ancillary No-Data with the previous CONTEXT of 10ms or more following the previous stream.

Afterwards, when the next stream can be recognized beforehand, the insertion of Ancillary No-Data with the next CONTEXT is recommended.

Otherwise, the insertions of Ancillary No-Data with next CONTEXT are not needed.

That is, the Ancillary No-Data with previous CONTEXT can be changed to the following stream directly.

And, when the transmission device does not have the capability of outputting the Ancillary No-Data with previous CONTEXT and the Ancillary No-Data with next CONTEXT, the transmission device can output MIDI No-Data or Empty packets or stop the stream output.

When the Empty packets is output to prevent losing the beginning of the following Content, it is preferable to add time stamp information in SYT.

11.3.1.5 Recommendation for receiver

1) Stereo products that receive multichannel streams with DBS >= 2 should reproduce the sound of the first two channels of the Data Block as Stereo Left and Right channels.

A.1.5 Stereo products which have no non-linear PCM decoder should reproduce no (muted) sound when they receive Validity Flag = '1' in IEC 60958 Conformant Data.

12. AM824 sequence adaptation layers

The transport mechanism using CIP may be used as an alternative transport layer for an existing data transmission protocol such as IEC 60958 and MIDI.

This adaptation layer definition defines only one-to-one mapping between an application data structure and an AM824 data structure and a procedure for transporting the application data only with a constant time shift.

The definition of the adaptation to CIP can be described and maintained by either organization responsible for the adaptation.

The adaptation layer definition described in this document provides only an alternative transport. The meaning of the data carried by the transport should be given in the original specification. Also, the transmission rate should be identical to that which is originally specified when the “non-identical to sampling frequency” indication flag is off.

The Adaptation Layer definition falls into two categories. One is generic that can be used in applications and does not define Application Specific Ancillary Data. Another is application specific that defines the structure of the Compound Data Block and Application Specific Ancillary Data.

12.1 General

12.1.1 IEC 60958 bitstream

All the information defined in IEC 60958 standard is mapped into this data format. Application, which uses IEC 60958 Conformant data shall follow IEC 60958 standard.

12.1.1.1 Sampling frequency in IEC 60958-3

In IEC 60958-3, Digital audio interface – Part 3 : Consumer applications [5], six new sampling frequencies of 22.05 kHz, 24 kHz, 88.2 kHz, 96 kHz, 176.4 kHz and 192 kHz are defined with bits 24-27 "sampling frequency" of channel status as shown in Table 12-1 – Sampling frequency in IEC 60958-3. All other combinations are reserved and shall not be used until further defined.

Table 12-1 – Sampling frequency in IEC 60958-3

State of bit 24 25 26 27	Sampling frequency
"0 0 0 0"	44.1 kHz
"1 0 0 0"	Sampling frequency not indicated
"0 1 0 0"	48 kHz
"1 1 0 0"	32 kHz
"0 0 1 0"	22.05 kHz
"0 1 1 0"	24 kHz
"0 0 0 1"	88.2 kHz
"0 1 0 1"	96 kHz
"0 0 1 1"	176.4 kHz
"0 1 1 1"	192 kHz

12.1.1.2 Original sampling frequency

Bits 36-39 are defined as “original sampling frequency” as shown in Table 12-2 – Original sampling frequency.

Table 12-2 – Original sampling frequency

State of bit 36 37 38 39	Original sampling frequency
"0 0 0 0"	Original sampling frequency not indicated
"1 0 0 0"	192 kHz
"0 1 0 0"	12 kHz
"1 1 0 0"	176.4 kHz
"0 0 1 0"	Reserved
"1 0 1 0"	96 kHz
"0 1 1 0"	8 kHz
"1 1 1 0"	88.2kHz
"0 0 0 1"	16 kHz
"1 0 0 1"	24kHz
"0 1 0 1"	11.025 kHz
"1 1 0 1"	22.05 kHz
"0 0 1 1"	32 kHz
"1 0 1 1"	48kHz
"0 1 1 1"	Reserved
"1 1 1 1"	44.1 kHz

12.1.1.3 Relation of sampling frequency and original sampling frequency

The sampling frequencies in IEC 60958-3 have relations of multiple of integer number as follows.

32 kHz line: (8 kHz, 16 kHz), 32 kHz
44.1 kHz line: (11.025 kHz), 22.05 kHz, 44.1 kHz, 88.2 kHz, 176.4 kHz
48 kHz line: (12 kHz), 24 kHz, 48 kHz, 96 kHz, 192 kHz

Sampling frequencies in parenthesis are defined only in original sampling frequency. Original sampling frequency is recorded in disc or transmitted by broadcasting and supplied from source device, e.g. player, tuner, etc.

12.1.1.4 Up or down sampling ratio

Original sampling frequency can be up-sampled or down-sampled. When up or down sampling ratio is defined, relation of sampling frequency and original sampling frequency is expressed using the following formula.

Sampling frequency = original sampling frequency ∗ up or down sampling ratio *(10)*

Table 12-3 – Up or down sampling ratio of 32 kHz line

Original sampling frequency	Sampling frequency
	32 kHz
8 kHz	4
16 kHz	2
32 kHz	1

Table 12-4 – Up or down sampling ratio of 44.1 kHz line

Original sampling frequency	Sampling frequency			
	22.05 kHz	44.1 kHz	88.2 kHz	176.4 kHz
11.025 kHz	2	4	8	16
22.05 kHz	1	2	4	8
44.1 kHz	1/2	1	2	4
88.2 kHz	1/4	1/2	1	2
176.4 kHz	1/8	1/4	1/2	1

Table 12-5 – Up or down sampling ratio of 48 kHz line

Original sampling frequency	Sampling frequency			
	24 kHz	48 kHz	96 kHz	192 kHz
12 kHz	2	4	8	16
24 kHz	1	2	4	8
48 kHz	1/2	1	2	4
96 kHz	1/4	1/2	1	2
192 kHz	1/8	1/4	1/2	1

12.1.1.5 Clock accuracy in IEC 60958-3

In IEC 60958-3, “11” in bits 28-29 of channel status is defined as “Interface frame rate not matched to sampling frequency”, shown in Table 12-6 – Clock accuracy in IEC 60958-3.

Table 12-6 – Clock accuracy in IEC 60958-3

State of bit 28 29	Clock accuracy
"0 0"	Level II
"1 0 "	Level I
"0 1"	Level III
"1 1"	Interface frame rate not matched to sampling frequency

12.1.1.6 High-speed transmission ratio and interface frame rate

High-speed transmission can be executed over IEC 60958, Digital audio interface. Original sampling frequency, high-speed transmission ratio and interface frame rate is expressed using the following formula.

Interface frame rate = original sampling frequency ∗ up or down sampling ratio ∗ high-speed transmission ratio *(11)*

Clock accuracy “11” in Table 12-6 – Clock accuracy in IEC 60958-3 means that high-speed transmission is executed. When Clock accuracy is “11”, there are two cases that sampling frequency is equal to original sampling frequency, and sampling frequency is not equal to original sampling frequency. The former case means that there is no up or down sampling process and the latter case means there is up or down sampling process. When Clock accuracy is “11”, Interface frame rate may be different from sampling frequency.

Clock accuracy “00”, “01” or “10” means that there is no high-speed transmission. When Clock accuracy is “00”, “01” or “10”, there are two cases that Sampling frequency is equal to Original sampling frequency, and Sampling frequency is not equal to Original sampling frequency. The latter case means there is up or down sampling process and the former case means that there is neither up or down sampling nor high-speed transmission. When Clock accuracy is “00”, “01” or “10”, interface frame rate is the same with sampling frequency.

These cases are illustrated in Table 12-7 – Cases for valid combination of channel status.

Table 12-7 – Cases for valid combination of channel status

Clock Accuracy	Original sampling frequency	Sampling frequency	Interface frame rate	Case
11	Original sampling frequency	Not equal to original sampling frequency	Not equal to sampling frequency	High-speed transmission and up or down sampling
11	Original sampling frequency	Equal to original sampling frequency	Not equal to sampling frequency	High-speed transmission
00, 01, 10	Original sampling frequency	Not equal to original sampling frequency	Equal to sampling frequency	Up or down sampling
00, 01, 10	Original sampling frequency	Equal to original sampling frequency	Equal to sampling frequency	Original

In Table 12-8 – Example for typical combination of source device and interface condition, some examples of cases are described.

Table 12-8 – Example for typical combination of source device and interface condition

Source device condition					Interface condition		
Original sampling frequency	Up or down sampling ratio	Sampling frequency	High-speed transmission ratio	Interface frame rate	Clock accuracy Bit 28, 29	Original sampling frequency Bit 36-39	Sampling frequency Bit 24-27
44.1 kHz	2	88.2 kHz	1	88.2 kHz	00,01,10	1111	0001
	1	44.1 kHz	1	44.1 kHz	00,01,10		0000
			2	88.2 kHz	11		
			4	176.4 kHz	11		
96 kHz	1	96 kHz	1	96 kHz	00,01,10	1010	0101
			2	192 kHz	11		
	1/2	48 kHz	1	48 kHz	00,01,10		0100
			2	96 kHz	11		
			4	192 kHz	11		
192 kHz	1	192 kHz	1	192 kHz	00,01,10	1000	0111
	1/2	96 kHz	1	96 kHz	00,01,10		0101
			2	192 kHz	11		
	1/4	48 kHz	1	48 kHz	00,01,10		0100
			2	96 kHz	11		
			4	192 kHz	11		

NOTE − if interface frame rate is equal to original sampling frequency, there may be up or down sampling process and high-speed transmission process. See marked (*) portion.

12.1.1.7 N-flag

N-flag is used with AM824 Data in general. Please refer to Clause 10.1 N-flag, 10.4 Command based rate control mode (FDF = 00001xxx2) and AV/C Command Set for Rate Control of Isochronous Data Flow 1.0 [R18].

When N-flag = 1, RATE CONTROL command with BASE CONFIGURE Subfunction can execute high-speed transmission of AM824 data over IEEE 1394 Bus and RATE CONTROL command with FLOW CONTROL Subfunction can execute flow control of AM824 data. This sub-clause describes relation between channel status coding in IEC 60958-3 and N-flag, SFC and SYT-INTERVAL in IEC 60958 Conformant data.

With introduction of IEC 60958-3 and Command based RATE CONTROL, the following cases may happen.

a) Real time transmission of 96 kHz (or 192 kHz) original sampling PCM signal over IEC 60958, *Digital audio interface*.
b) Real time transmission of up-sampled 48 kHz original sampling PCM signal by 96 kHz (or 192 kHz) sampling frequency over IEC 60958, *Digital audio interface*.

c) High-speed transmission of 48 kHz original sampling PCM signal with 96 kHz (or 192 kHz) sampling frequency over IEC 60958, *Digital audio interface*.
d) Double (or four times) high-speed transmission of 48 kHz original sampling PCM signal with RATE CONTROL command with BASE CONFIGURE Subfunction over IEEE 1394 Bus.

When IEC 60958 signals of a), b) and c) are transmitted over IEEE 1394 Bus with IEC 60958 Conformant mode, some mechanisms are necessary to distinguish signals of a), b), c) and d) on isochronous mode.

For cases of a), b) and c), IEC 60958-3 specifies new codes in clock accuracy (see Table 12-6 – Clock accuracy in IEC 60958-3) and original sampling frequency (see Table 12-2 – Original sampling frequency) in channel status. For case of d), N-flag is set to “1”.

When N-flag = 0, values of SYT_INTERVAL and *Nominal_Sampling_Frequency* are described in Table 10-1 – Default SFC table for FDF = 0000 $0xxx_2$. When N-flag = 1, values of SYT_INTERVAL and *Nominal_Sampling_Frequency* are described in Table 10-3 – Default SFC table for FDF = 0000 $1xxx_2$. For IEC 60958 Conformant data, the following rules for SYT_INTERVAL and *Nominal_Sampling_Frequency* are applied.

1) When N-flag = 0, value of *Nominal_Sampling_Frequency* for IEC 60958 Conformant data is set according to interface frame rate and value of SYT_INTERVAL is set corresponding to the *Nominal_Sampling_Frequency* in Table 10-1 – Default SFC table for FDF = 0000 $0xxx_2$.

2) When N-flag = 1, value of *Nominal_Sampling_Frequency* for IEC 60958 Conformant data is set according to sampling frequency coded in Bit 24-27 of channel status of IEC 60958-3 format.

3) When N-flag = 1 and RATE CONTROL command with BASE CONFIGURE Subfunction is executed, value of SYT_INTERVAL is set to *n* multiplexed by SYT_INTERVAL value corresponding to the *Nominal_Sampling_Frequency* in Table 10-3 – Default SFC table for FDF = 0000 $1xxx_2$.

4) When N-flag = 1 and RATE CONTROL command with BASE CONFIGURE Subfunction is executed, clock Accuracy, bit 28-29 of channel Status, of IEC 60958-3 is set to ‘11’.

5) When N-flag = 1 and RATE CONTROL command with FLOW CONTROL Subfunction is executed, value of SYT_INTERVAL is set corresponding to the *Nominal_Sampling_Frequency* in Table 10-3 – Default SFC table for FDF = 0000 $1xxx_2$.

6) When N-flag = 1 and RATE CONTROL command with FLOW CONTROL Subfunction is executed, clock Accuracy, bit 28-29 of channel Status, of IEC 60958-3 is set to ‘00’, ‘01’ or ‘10’.

Table 12-9 – Relation of values in IEC 60958-3 and A/M Protocol

	IEC 60958-3			Interface frame rate	A/M Protocol		
	Bits 36-39 Original sampling frequency	Bits 24-27 Sampling frequency	Bits 28 29 Clock accuracy		N-flag	SFC	SYT-INTERVAL
Case a) Original IEC 60958-3	96 kHz (192 kHz)	96 kHz (192 kHz)	00,01,10	96 kHz (192 kHz)	–	–	–
Case b) Up sampling IEC 60958-3	48 kHz	96 kHz (192 kHz)	00,01,10	96 kHz (192 kHz)	–	–	–
Case c) High-speed IEC 60958-3	48 kHz	48 kHz	11	96 kHz (192 kHz)	–	–	–
Case a) Original with A/M Protocol	96 kHz (192 kHz)	96 kHz (192 kHz)	00,01,10	96 kHz (192 kHz)	0	96 kHz (192 kHz)	16 (32)
Case b) Up sampling with A/M Protocol	48 kHz	96 kHz (192 kHz)	00,01,10	96 kHz (192 kHz)	0	96 kHz (192 kHz)	16 (32)
Case c) High-speed with A/M Protocol	48 kHz	48 kHz	11	96 kHz (192 kHz)	0	96 kHz (192 kHz)	16 (32)
Case d) Rate control with A/M Protocol	48 kHz	48 kHz	11	96 kHz (192 kHz)	1	48 kHz	$8 * n$ ($n = 2, 4$)
FLOW CONTROL with A/M Protocol	96 kHz (192 kHz)	96 kHz (192 kHz)	00,01,10	96 kHz (192 kHz)	1	96 kHz (192 kHz)	16 (32)

12.1.2 One Bit Audio

In this clause, the format of One Bit Audio is described.

12.1.2.1 One Bit Audio (plain)

The data of the One Bit Audio (LABEL = 50_{16}- 51_{16}) has one bit length data stream, and can be directly played back through the analog low pass filter bit by bit (MSB First). The data stream is packed in 24-bit data fields of an AM824 quadlet with MSB First per audio channel.

The sampling frequency of the One Bit Audio (LABEL = 50_{16},51_{16},58_{16}) is defined in Table 12-10 – Sampling frequency definition of One Bit Audio with its own SFC table.

Table 12-10 – Sampling frequency definition of One Bit Audio

Value of SFC	SYT_INTERVAL	Sampling Frequency
00	16	2.048MHz
01	16	2.8224MHz
02	32	3.072MHz
03	32	5.6448MHz
04	64	6.144MHz
05	64	11.2896MHz
06	128	12.288MHz
07	- reserved -	- reserved -

The TRANSFER_DELAY for Blocking Transmission, in the case of DEFAULT_TRANSFER_DELAY = 479.17µsec = (354.17 + 125)µsec, corresponds to the Table 12-10 – Sampling frequency definition of One Bit Audio as given in Table 12-11 – TRANSFER_DELAY for blocking transmission in the case of the One Bit Audio.

Table 12-11 – TRANSFER_DELAY for blocking transmission in the case of the One Bit Audio

Value	TRANSFER_DELAY
00_{10}	479.17 + 187.50 = 666.67 [μ sec]
01_{10}	479.17 + 136.10 = 615.27 [μ sec]
02_{10}	479.17 + 250.00 = 729.17 [μ sec]
03_{10}	479.17 + 136.10 = 615.27 [μ sec]
04_{10}	479.17 + 250.00 = 729.17 [μ sec]
05_{10}	479.17 + 136.10 = 615.27 [μ sec]
06_{10}	479.17 + 250.00 = 729.17 [μ sec]
07_{10}	- reserved -

The One Bit Audio (LABEL = 50_{16}- 51_{16}) can transmit Multi-Channel Cluster. Each AM824 quadlet carries the data for one channel of the cluster. Two AM824 LABELs are used to indicate the Start and Continuation of the data in the cluster.

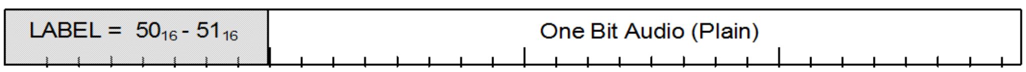

Figure 12-1 – Generic One Bit Audio quadlet

The channel number shall start with No.1 and be sequential:

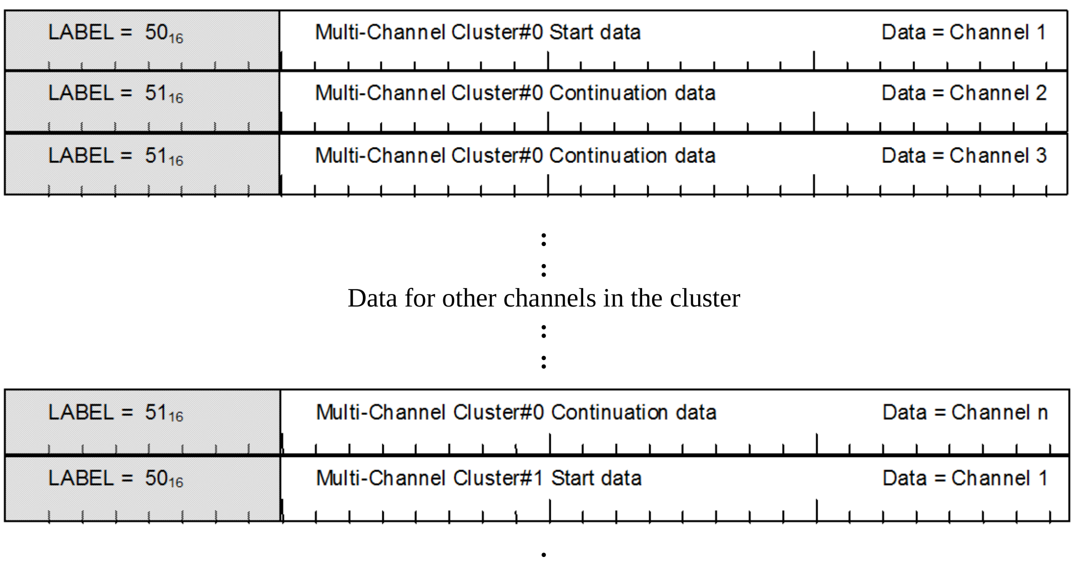

Figure 12-2 – Generic One Bit Audio quadlet sequence

12.1.2.2 One Bit Audio (encoded)

The data of the One Bit Audio (Encoded) is the encoded data stream.

12.1.2.2.1 DST

DST (Direct Stream Transfer) is the loss-less coding technique used for One Bit Audio in SACD and is defined in the [B1] Part 2.

The encoded data stream is packed in 24-bit data fields of AM824 Data with MSB First.

For decoding the stream, SACD Ancillary Data is needed. DST supports multi channel One Bit Audio and carries each data stream in one mixed stream.

DST encodes the One Bit Audio data stream Frame by Frame. The Frame is defined in [B1] Part 2.

The sampling frequency of the DST is defined in Table 12-10 – Sampling frequency definition of One Bit Audio with its own SFC table.

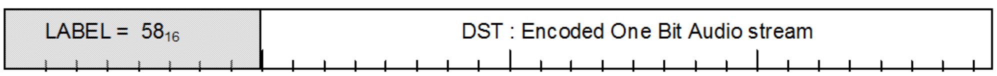

Figure 12-3 – One Bit Audio DST encoded quadlet

12.1.2.3 High speed transfer for One Bit Audio

As far as One Bit Audio (LABEL = 50_{16}, 51_{16}, 58_{16}), the transfer frequency and SYT_INTERVAL for the high speed AM824-data transfer are defined depending on the speed as shown in Table 12-12 – SFC definition of One Bit Audio for high speed AM824-data transfer if N-flag in the FDF is 1. In this table, an integer value of *n* (>1) indicates the number of times faster than normal speed.

Table 12-12 – SFC definition of One Bit Audio for high speed AM824-data transfer

Value of SFC	*Nominal_Sampling _Frequency*	SYT_INTERVAL	*Sampling_Frequency*
0	2.048 MHz	16 * *n*	2.048 MHz * *n*
1	2.8224 MHz	16 * *n*	2.8224 MHz * *n*
2	3.072 MHz	32 * *n*	3.072 MHz * *n*
3	5.6448 MHz	32 * *n*	5.6448 MHz * *n*
4	6.144 MHz	64 * *n*	6.144 MHz * *n*
5	11.2896 MHz	64 * *n*	11.2896 MHz * *n*
6	12.288 MHz	128 * *n*	12.288 MHz * *n*
7	- reserved -	- reserved -	- reserved -

The DBS of an event is independent of the transfer speed.

12.1.3 Non-linear audio data stream

Any non-linear audio data carried by an IEC 61937 bitstream can be transmitted by using the IEC 60958 Conformant data sequence.

12.1.4 MIDI data stream

Any modification or enhancement is prohibited on this adaptation layer although increase of transmission rate for instance can be easily done. The specification that uses this adaptation layer is given in [R14].

This specification restricts the packetization of MIDI data stream so that a single MIDI Conformant sequence can carry multiple MIDI data streams by multiplexing. The MIDI Conformant data defines MULTIPLEX_NUMBER = 8.

NOTE – The Default MULTIPLEX_NUMBER for MIDI Conformant AM824 types may be incompatible with some applications conforming to IEC 61883-6 [R4].

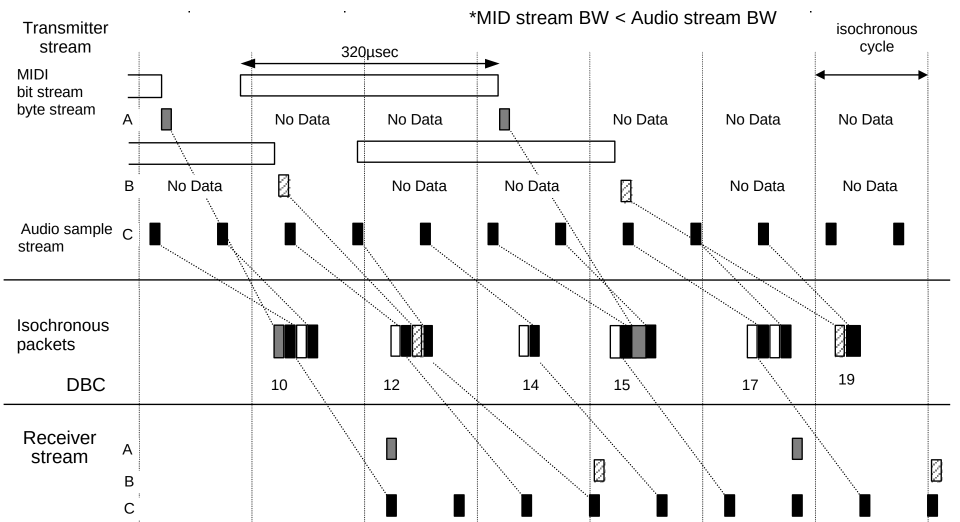

Figure 12-4 – Multiplexing of MIDI data streams (informative)

NOTE – Figure 12-4 – Multiplexing of MIDI data streams (informative) shows how two MIDI data streams, which should flow in different MIDI cables, are multiplexed in a single MIDI Conformant sequence with an audio stream. This figure is intended to give only the sequence multiplexing scheme. The parameters of this example such as the number of multiplexed sequences and the audio sampling rate were chosen so that the figure would be readable. Consequently, not all the parameters are valid for this specification and its predecessor.

12.1.5 SMPTE time code and sample count

SMPTE time code and sample count transmission are defined in a separate document [R16]

12.1.6 High Precision and Double Precision Multi-bit Linear Audio

Double Precision uses the LABEL from 60_{16} to 61_{16}.

12.1.6.1 High Precision specific Ancillary Data

This clause specifies private header data that are carried by High Precision specific Ancillary Data (informative).

This Ancillary Data is transmitted at every data block.

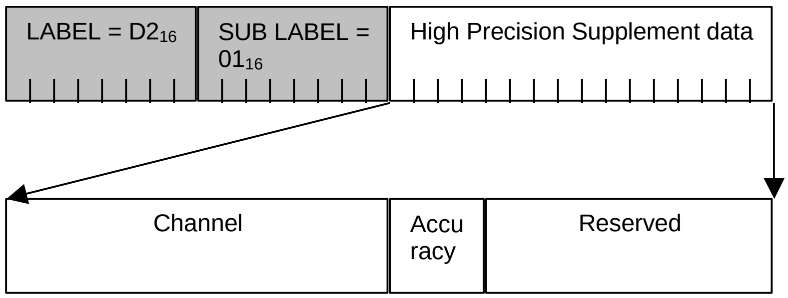

Figure 12-5 – High Precision First Ancillary Data

Table 12-13 – Channel definition

Value	Description
0000 0000_2	1 Channel
0000 0001_2	2 Channel
0000 0010_2	3 Channel
…	…
1111 1110_2	255 Channel
1111 1111_2	256 Channel

Table 12-14 – Accuracy definition

Value	Description
00_2	16 bit slot (Lower 8 bit = 0)
01_2	20 bit slot (Lower 4 bit = 0)
10_2	24 bit slot
11_2	- reserved -

With the combination of Accuracy and Num. (slot number), any PCM Audio data with sample word length up to 192 bits can be transmitted by High Precision Multi-bit linear audio. There is a wide redundancy, for example 64 bit sample word can be transmitted with 3 slots of 24 bit slot (Acc = 10_2) or 4 slots of 16 bit slot (Acc = 00_2). To eliminate hardware complexity of decoder side, the following implementation rules are strongly recommended.

(1) Sample word should be limited to 32, 40, 48, 64, 80, 96, 128, 160 and 192 bit length.

(2) Number of slots should be limited to 2, 4 and 8.

Accuracy for the above sample words should be specified in Table 12-15 – Recommended rules.

Table 12-15 – Recommended rules

Sample word length	Accuracy		Number of slots
	Value	Slot length	
32 bits	00_2	16 bits	2
40 bits	01_2	20 bits	2
48 bits	10_2	24 bits	2
64 bits	00_2	16 bits	4
80 bits	01_2	20 bits	4
96 bits	10_2	24 bits	4
128 bits	00_2	16 bits	8
160 bits	01_2	20 bits	8
192 bits	10_2	24 bits	8

When a source device sends its own data or auxiliary information to sink device with High Precision mode, its original data and/or ancillary data can be transmitted between the High Precision First Ancillary Data and High Precision Mutlti-bit Linear Audio data.

For example, IEC 60958 Conformant data can be transmitted between the High Precision First Ancillary Data and High Precision Mutlti-bit Linear Audio data. For other applications, Common and Application Specific Ancillary Data can be transmitted between the High Precision First Ancillary Data and High Precision Mutlti-bit Linear Audio data. Refer to 8.1.9.1 Common and 8.1.10 Application Specific Ancillary Data and each application sections.

$D2_{16}$	01_{16}	Ancillary Data
$3X_{16}$	IEC 60958 Conformant Left Channel	
$0X_{16}$	IEC 60958 Conformant Right Channel	
60_{16}	Upper 24 bit Data for Channel 1	
61_{16}	Lower 24 bit Data for Channel	
60_{16}	Upper 24 bit Data for Channel 2	
61_{16}	Lower 24 bit Data for Channel 2	

Figure 12-6 – IEC 60958 Conformant data with High Precision data

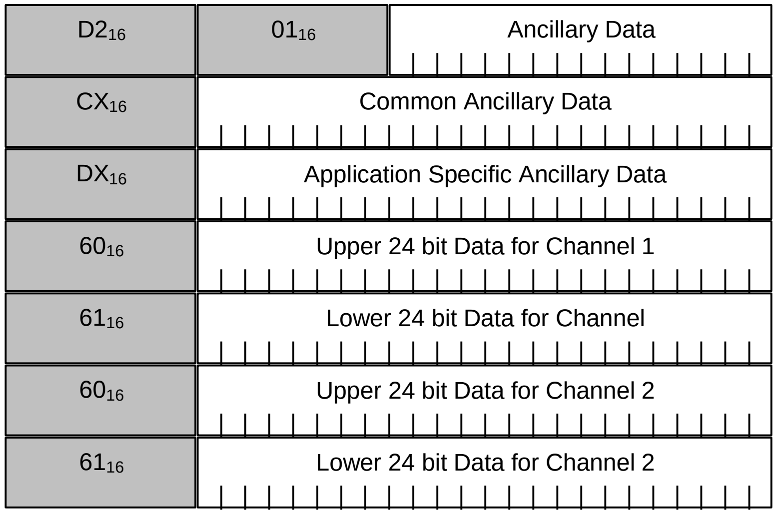

Figure 12-7 – Common and Application Specific Ancillary data with High Precision data

This Ancillary Data is optional and its definition is reserved.

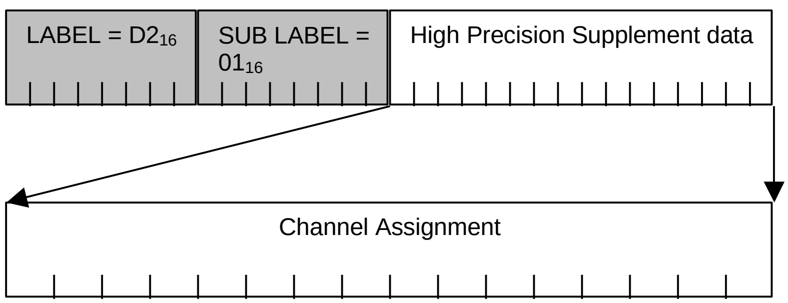

Figure 12-8 – High Precision Channel Assignment Ancillary Data

Table 12-16 – Channel Assignment definition

Channel Assignment	Description
0000 0000 0000 0000$_2$	- reserved -
0000 0000 0000 0001$_2$	
0000 0000 0000 0010$_2$	
…	
1111 1111 1111 1110$_2$	
1111 1111 1111 1111$_2$	

12.1.6.2 Example of High Precision stream

Figure 12-9 – Example of High Precision data shows a 2 channel 128 bit sample word High Precision stream carried over the Serial Bus. Here, the lower 8 bits are set to "0". High Precision Ancillary Data is immediately followed by the data block.

0 MSB — Quadlets sequence — 31 LSB

$D2_{16}$	01_{16}	Ancillary Data
60_{16}	1st 16 bit Data for Channel 1	
61_{16}	2nd 16 bit Data for Channel 1	
62_{16}	3rd 16 bit Data for Channel 1	
63_{16}	4th 16 bit Data for Channel 1	
64_{16}	5th 16 bit Data for Channel 1	
65_{16}	6th 16 bit Data for Channel 1	
66_{16}	7th 16 bit Data for Channel 1	
67_{16}	8th 16 bit Data for Channel 1	
60_{16}	1st 16 bit Data for Channel 2	
61_{16}	2nd 16 bit Data for Channel 2	
62_{16}	3rd 16 bit Data for Channel 2	
63_{16}	4th 16 bit Data for Channel 2	
64_{16}	5th 16 bit Data for Channel 2	
65_{16}	6th 16 bit Data for Channel 2	
66_{16}	7th 16 bit Data for Channel 2	
67_{16}	8th 16 bit Data for Channel 2	

Figure 12-9 – Example of High Precision data

12.1.6.3 Example of Double Precision stream

The Figure 12-10 – Example of Double Precision data shows a 6 channel 48 bit sample word Double Precision stream carried over the Serial Bus.

0 MSB — Quadlets sequence — 31 LSB

$D2_{16}$	01_{16}	Ancillary Data
60_{16}	Upper 24 bit Data for Channel 1	
61_{16}	Lower 24 bit Data for Channel 1	
60_{16}	Upper 24 bit Data for Channel 2	
61_{16}	Lower 24 bit Data for Channel 2	
60_{16}	Upper 24 bit Data for Channel 3	
61_{16}	Lower 24 bit Data for Channel 3	
60_{16}	Upper 24 bit Data for Channel 4	
61_{16}	Upper 24 bit Data for Channel 4	
60_{16}	Lower 24 bit Data for Channel 5	
61_{16}	Upper 24 bit Data for Channel 5	
60_{16}	Lower 24 bit Data for Channel 6	
61_{16}	Upper 24 bit Data for Channel 6	

Figure 12-10 – Example of Double Precision data

12.1.6.4 Example of Double Precision Compound stream

The Figure 12-11 – Example of Double Precision Compound data shows a 4 channel 48 bit sample word Double Precision Compound stream carried over the Serial Bus.

	0 MSB	Quadlets sequence	31 LSB
Data Block [m]	$D2_{16}$	01_{16}	Ancillary Data
	60_{16}	Upper 24 bit Data for Channel 1	
	61_{16}	Lower 24 bit Data for Channel 1	
	60_{16}	Upper 24 bit Data for Channel 2	
	61_{16}	Lower 24 bit Data for Channel 2	
	60_{16}	Upper 24 bit Data for Channel 3	
	61_{16}	Lower 24 bit Data for Channel 3	
	60_{16}	Upper 24 bit Data for Channel 4	
	61_{16}	Lower 24 bit Data for Channel 4	
Data Block [m+1]	$D2_{16}$	01_{16}	Ancillary Data
	60_{16}	Upper 24 bit Data for Channel 1	
	61_{16}	Lower 24 bit Data for Channel 1	
	60_{16}	Upper 24 bit Data for Channel 2	
	61_{16}	Lower 24 bit Data for Channel 2	
	60_{16}	Upper 24 bit Data for Channel 3	
	61_{16}	Lower 24 bit Data for Channel 3	
	60_{16}	Upper 24 bit Data for Channel 4	
	61_{16}	Lower 24 bit Data for Channel 4	

Figure 12-11 – Example of Double Precision Compound data

12.2 DVD-Audio

The compound data for DVD-Audio consists of Multi-bit Linear Audio data, Common Ancillary and DVD-Audio Specific Ancillary Data.

12.2.1 Multi-bit linear audio data

DVD-Audio data use the LABEL from 48_{16} to $4F_{16}$ of Multi-bit Linear Audio and use ASI2 for scaleable contents.

Table 12-17 – ASI2 definition for DVD-Audio

Value	Description
00_2	24 bits
01_2	20 bits
10_2	16 bits
11_2	Previous Sample Word Data Hold

12.2.2 DVD-Audio Specific Ancillary Data

This clause specifies private header data that are carried by DVD-Audio Specific Ancillary Data (informative).

Table 12-18 – DVD-Audio Specific Ancillary Data

LABEL	SUB LABEL	Description
$D0_{16}$	01_{16}	Data transmitted at every data block
	02_{16}	Data transmitted at starting point
	$C0_{16}$	Audio CCI
	$C1_{16}$	ISRC

12.2.2.1 Data transmitted at starting point

This Ancillary Data is used at the starting point of audio data when performing play start or search for a track number.

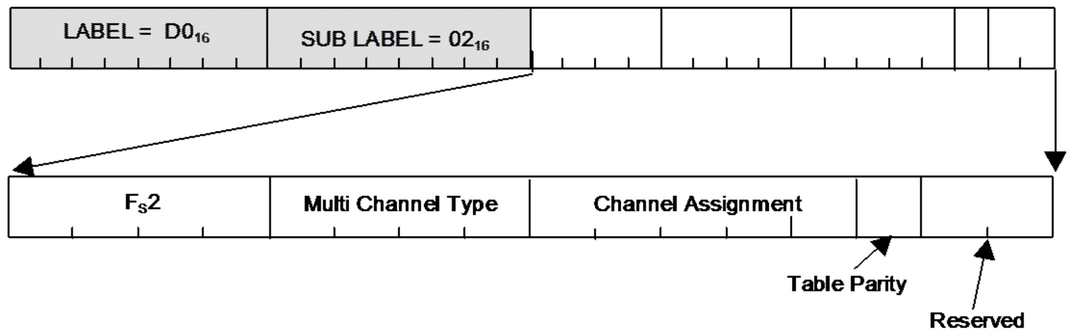

Figure 12-12 – Data transmitted at data starting point

Table 12-19 – Data transmitted at starting point

Data	Bits	Description
Fs2	4	Sampling Frequency Group2
Multi Channel Type	4	Fs, Bit combination table
Channel Assignment	5	Channel combination of Group1 and 2
Table Parity	1	Table Parity of audio data

12.2.2.2 Data transmitted at every data block

This Ancillary Data is transmitted at every data block.

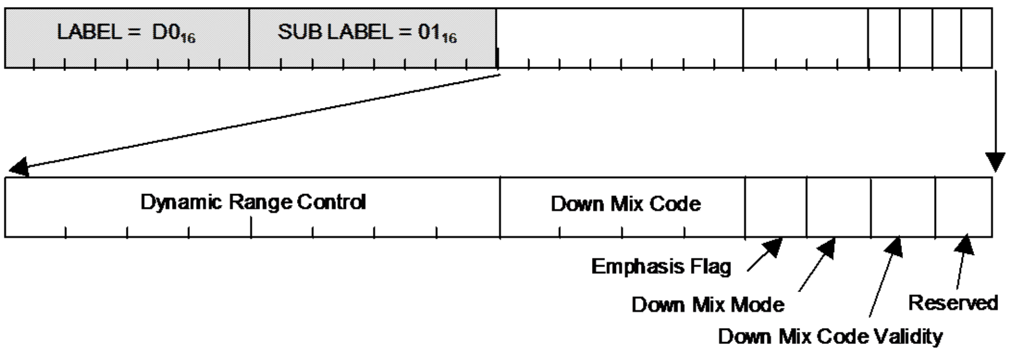

Figure 12-13 – Data transmitted at every data block

Table 12-20 – Data transmitted at every data block

Data	Bits	Description
Dynamic Range Control	8	Adaptive compression coefficient
Down Mix Code	4	Down Mix Table number
Emphasis Flag	1	Enhances on or off
Down Mix Mode	1	Down Mix permission
Down Mix Code Validity	1	Down Mix Code validity

12.2.3 Data for CCI

SUB LABEL C0$_{16}$ is for CCI.

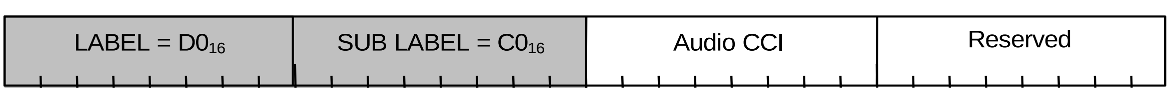

Figure 12-14 – Ancillary Data for CCI

NOTE: Audio CCI is Copy Control Information for Audio

12.2.4 Data for ISRC

SUB LABEL C1$_{16}$ is for ISRC.

LABEL = $D0_{16}$	SUB LABEL = $C1_{16}$	ISRC

Figure 12-15 – Ancillary Data for ISRC

12.2.5 Example of DVD-Audio stream

Figure 12-16 – Basic data block of DVD-Audio stream illustrates a basic data block of DVD-Audio stream carried over the 1394 Bus in the case of six channels.

$D0_{16}$	01_{16}	Ancillary Data
$D0_{16}$	02_{16}	Ancillary Data
48_{16}	Data Channel 1	
48_{16}	Data Channel 2	
48_{16}	Data Channel 3	
48_{16}	Data Channel 4	
48_{16}	Data Channel 5	
48_{16}	Data Channel 6	

Figure 12-16 – Basic data block of DVD-Audio stream

Data on the disc is organized into a series of blocks. The data for each channel is packed into one block. Each data block should be ordered by increasing channel number. The data block immediately follows DVD-Audio ancillary data. The first ancillary data is “the data transmitted at every data block,” and the second ancillary data is “the data transmitted at the data starting point” or “ Table Parity” or “ DMCT (Down Mix Coefficient Table)” or something similar.

Figure 12-17 – Example of DVD-Audio data illustrates an example of DVD-Audio data stream that carries scaleable contents of DVD-Audio. In this case, sampling frequency and sample word length may be different between front channels and rear channels, and in the second data block, previous data hold of ASI2 of DVD-Audio is used.

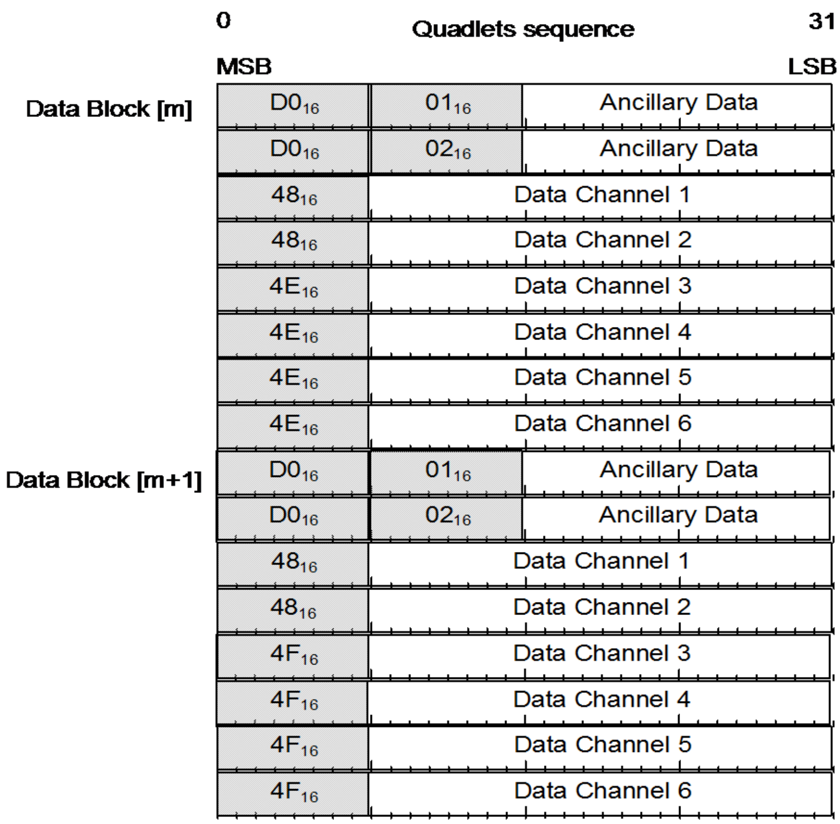

Figure 12-17 – Example of DVD-Audio data

12.3 SACD

The Data Block for SACD consists of One Bit Audio data, Common Ancillary and SACD specific Ancillary Data.

12.3.1 SACD Ancillary Data

The SACD player transmits SACD Ancillary Data at the starting point of every Frame. The Frame is defined in the [B1]. The SACD Ancillary Data contains the information about the data within the Frame.

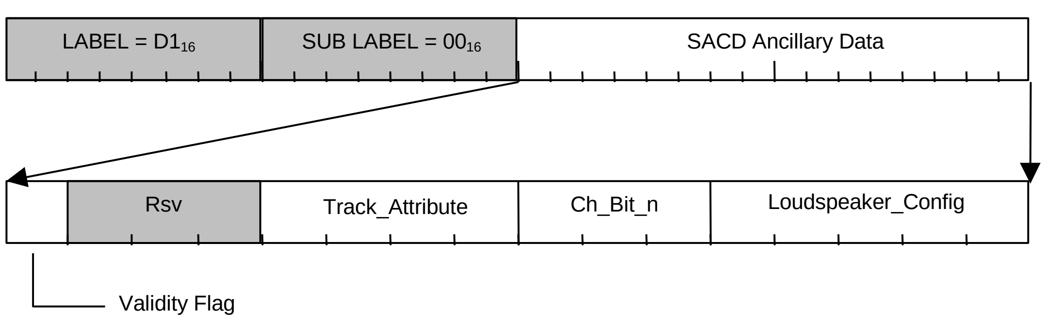

Figure 12-18 – SACD Ancillary Data

Table 12-21 – data information (informative)

Data	Bits	Description
Validity Flag	1	Valid or Not valid
Track_Attribute	4	Copy Control Information
Ch_Bit_n	3	Number of channels
Loudspeaker_Config	5	Loudspeaker set-up

The Validity Flag shows the validity of the data within the Frame.

If a disc read error occurs, the SACD player shall replace the error data with safe data, such as a mute signal, and set the Validity Flag to 1_2.

Table 12-22 – Validity flag definition

Value(binary)	Description
0_2	Valid
1_2	Not valid

Rsv is the reserved area and the default value is 000_2.

The Track_Attribute shows copy control information dedicated to Super Audio CD, and is defined in [B1].This information shall be copied from the Super Audio CD track by track.

The Ch_Bit_n shows the total number of channels, and is defined in. This information shall be copied from the Super Audio CD Frame by Frame.

The Loudspeaker_Config shows the loudspeaker set-up, and is defined in [B1]. This information shall be copied from Super Audio CD track by track.

12.3.2 SACD Supplementary Data

SACD Supplementary Data is a synchronized stream along with the Audio data from the SACD. It has several data lengths as defined in the [B1] . Audio Data and Supplementary Data are synchronized on a Frame by Frame basis.

For decoding the stream, SACD Ancillary Data is needed.

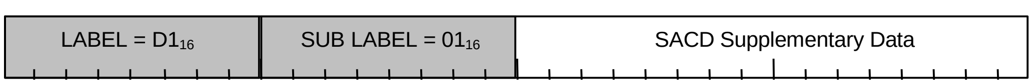

Figure 12-19 – SACD Supplementary Data

12.3.3 SACD Track_Mode&Flags Data

SACD Track_Mode&Flags Data consists of "Track_Mode" (1byte) and Track Flags (1byte) defined in the [B1]. The relationship between SACD Track_Mode&Flags,"Track_Mode" and "Track_Flags" is described below.

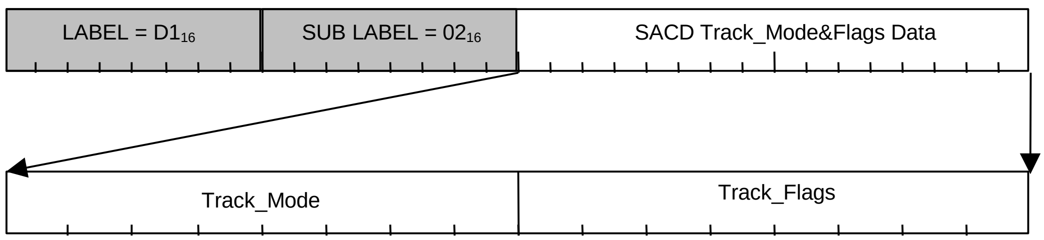

Figure 12-20 – SACD Track_Mode&Flags Data

12.3.4 SACD Track_Copy_Management Data

SACD Track_Copy_Management Data consists of three AM824 data quadlets, and shows the "Track_Copy_Management" defined in the [B1]. The data of the "Track_Copy_Management" (6bytes) is divided into three data fields (Part1, 2, 3) of the AM824 quadlets (AM824 LABEL=D1$_{16}$: SUB LABEL=10$_{16}$, 11$_{16}$, 12$_{16}$) in sequence.

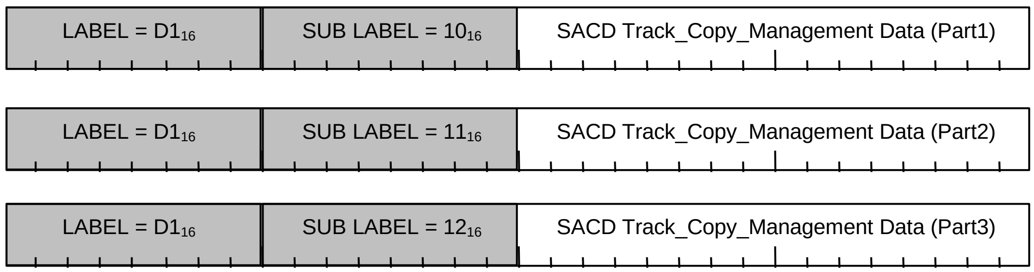

Figure 12-21 – SACD Track_Copy_Management Data

12.3.5 Example of SACD streams (informative)

Figure 12-22 – Example of SACD Stream in the case of six channels illustrates a typical multi-channel Plain One Bit Audio stream carried over the 1394 Bus from SACD for the case where the value of SFC in FDF is 001$_2$. The data on the disc is organized into a series of frames, with 75 frames for each second of audio. Each frame contains a total of 1568 * 3 bytes of Audio Cluster Data per channel. Quadlets in a Data Block are organized according to the "Order Rule", so that the order is Ancillary Data first, Multi-Channel Cluster data next, and an Ancillary No-Data with CONTEXT = CF$_{16}$ last.

The SACD Ancillary Data starts and is followed by the first group of Multi-Channel Cluster data. In this example, the first quadlet contains the Ancillary Data for the whole of Frame #0. If, for example, there is a disc error, the SACD player sets the Validity Flag in the Ancillary Data for this Frame (Frame #0) which remains valid until the next SACD Ancillary Data (Frame #1). This also applies to the Track_Attribute, Ch_Bit_n and Loudspeaker_Config contained in the Ancillary Data for Frame #0.

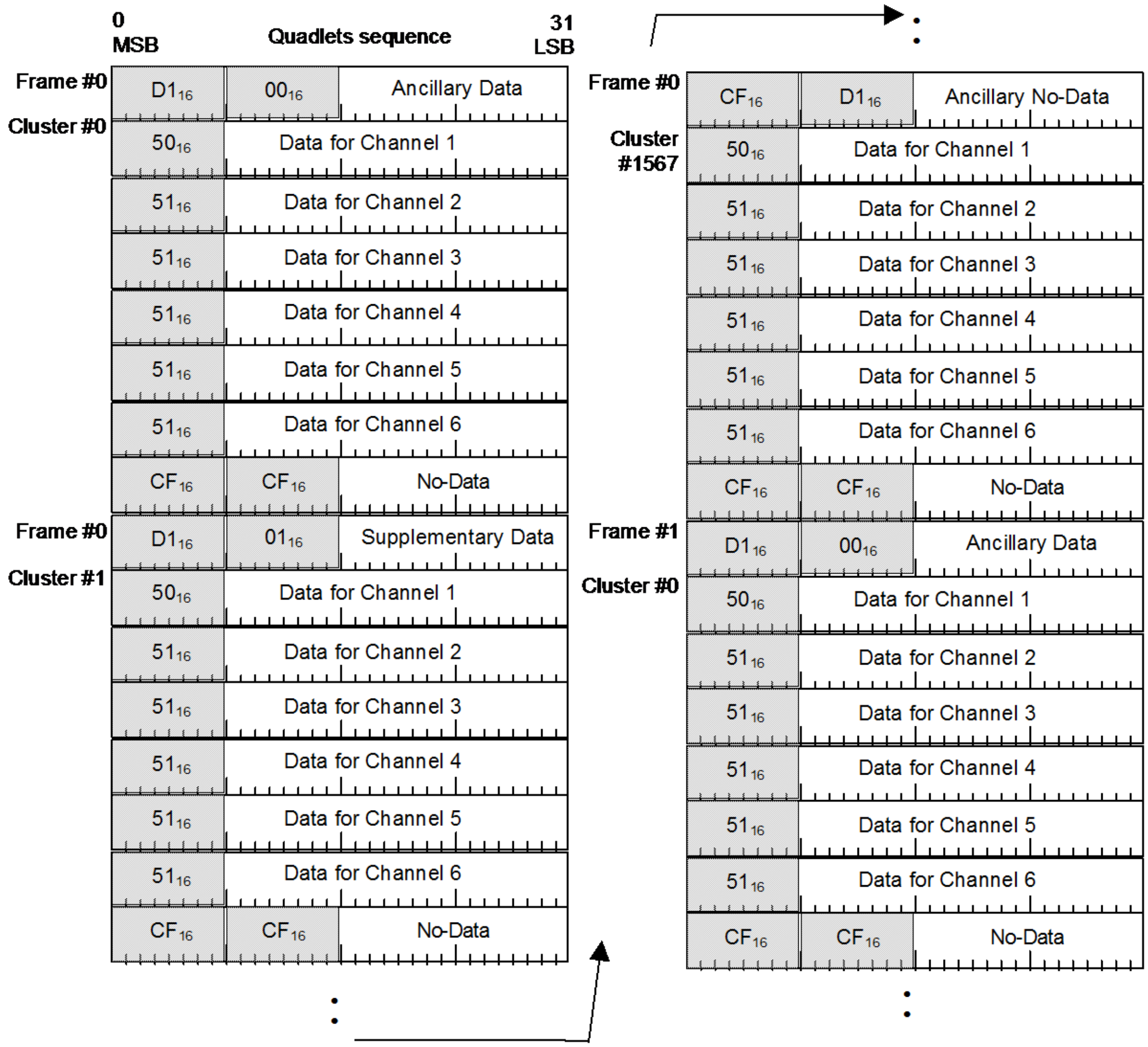

Figure 12-22 – Example of SACD Stream in the case of six channels

In the example of Figure 12-22 – Example of SACD Stream in the case of six channels, there are six channels in the Multi-Channel Cluster, so an Ancillary No-Data with CONTEXT = CF is added to the last of the cluster data so that the total numbers of quadlets in the block is kept even, therefore the DBS = 8. The SACD Supplementary Data is transmitted at the same location as the SACD Ancillary Data (after the SACD Ancillary Data has already been transmitted). After all the SACD Supplementary Data has been transmitted, an Ancillary No-Data or other Ancillary Data quadlet may be put in the same location.

Figure 12-23 – Example of SACD Stream in the case of five channels shows five channel cases. Here, Ancillary No-Data with CONTEXT = CF is not required, and the DBS = 6.

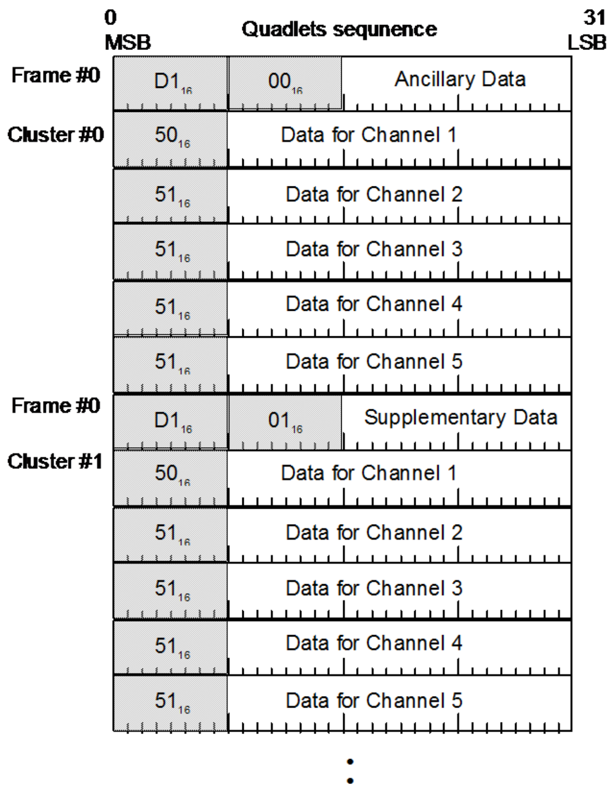

Figure 12-23 – Example of SACD Stream in the case of five channels

12.4 Blu-ray Disc

The compound data for Blu-ray Disc consists of Multi-bit Linear Audio data, Common Ancillary and Blu-ray Disc Specific Ancillary Data.

12.4.1 Structure of Sample Word for Audio Transmission

There are eight Sample Word in one audio sample.

LABEL	Data Channel 1 (Left channel)
LABEL	Data Channel 2 (Right channel)
LABEL	Data Channel 3 (low frequency effects channel)
LABEL	Data Channel 4 (Centre channel)
LABEL	Data Channel 5 (Left Surround channel)
LABEL	Data Channel 6 (Right Surround channel)
LABEL	Data Channel 7 (Rear surround left channel)
LABEL	Data Channel 8 (Rear surround right channel)

Figure 12-24 – Basic data block of Blu-ray Disc

Channel layout is fixed.

The transmitter shall set 000000_{16} on the audio data of MBLA data in the case of non existing channel.

Valid combinations of Sample Word are determined by the permitted channel allocations as defined in [B5].

12.4.2 Multi-bit linear audio data

Blu-ray Disc data use the LABEL from 48_{16} to $4A_{16}$ of Multi-bit Linear Audio.

Table 12-23 – ASI1 definition for Blu-ray Disc

Value	Description
00_2	-
01_2	- reserved -
10_2	Ordinary LPCM FS shown by SFC
11_2	- reserved -

Table 12-24 – ASI2 definition for Blu-ray Disc

Value	Description
00_2	24bits
01_2	20bits
10_2	16bits
11_2	- reserved -

12.4.3 Blu-ray Disc Specific Ancillary Data

The clause specifies private header data that are carried by Blu-ray Disc Specific Ancillary Data.

Table 12-25 – Blu-ray Disc Specific Ancillary Data

LABEL	SUB LABEL	Description
$D3_{16}$	01_{16}	Data transmitted at every data block
	$C0_{16}$	CCI

The transmission device shall execute stream change method if Ancillary Data is changed except when SUB LABEL is $C0_{16}$.

12.4.4 Data transmitted at every data block

This Ancillary Data is transmitted at every data block.

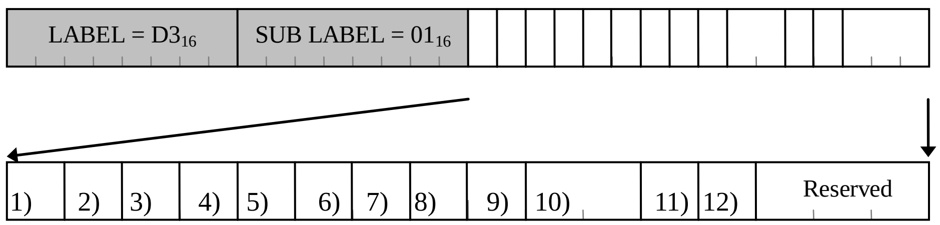

Figure 12-25 – Data transmitted at every data block

Table 12-26 – Data transmitted at every data block

Data	Bits	Description
1) reserved	1	- reserved -
2) L channel	1	L (Left) channel data exist or not exist
3) R channel	1	R (Right) channel data exist or not exist
4) lfe channel	1	Lfe (low frequency effects) channel data exist or not exist
5) C channel	1	C (Centre) channel data exist or not exist
6) LS channel	1	LS (Left Surround) channel data exist or not exist
7) RS channel	1	RS (Right Surround) channel data exist or not exist
8) Rls channel	1	Rls (Rear surround left) channel data exist or not exist
9) Rrs channel	1	Rrs (Rear surround right) channel data exist or not exist
10) L/R ch identifier	2	L (Left)/R (Right) channel identifier defined
11) C ch identifier	1	C (Centre) channel identifier defined
12) LS/RS ch identifier	1	LS (Left Surround)/RS (Right Surround) channel identifier defined

The L channel shows whether L (Left) channel data is existed or not.

Table 12-27 – L channel definition

Value	Description
0_2	L (Left) channel data is not existed
1_2	L (Left) channel data is existed

The R channel shows whether R (Right) channel data is existed or not.

Table 12-28 – R channel definition

Value	Description
0_2	R (Right) channel data is not existed
1_2	R (Right) channel data is existed

The lfe channel shows whether lfe (low frequency effects) channel data is existed or not.

Table 12-29 – lfe channel definition

Value	Description
0_2	lfe (low frequency effects) channel data is not existed
1_2	lfe (low frequency effects) channel data is existed

The C channel shows whether C (Centre) channel data is existed or not.

Table 12-30 – C channel definition

Value	Description
0_2	C (Centre) channel data is not existed
1_2	C (Centre) channel data is existed

The LS channel shows whether LS (Left Surround) channel data is existed or not.

Table 12-31 – LS channel definition

Value	Description
0_2	LS (Left Surround) channel data is not existed
1_2	LS (Left Surround) channel data is existe

The RS channel shows whether RS (Right Surround) channel data is existed or not.

Table 12-32 – RS channel definition

Value	Description
0_2	RS (Right Surround) channel data is not existed
1_2	RS (Right Surround) channel data is existed

The Rls channel shows whether Rls (Rear surround left) channel data is existed or not.

Table 12-33 – Rls channel definition

Value	Description
0_2	Rls (Rear surround left) channel data is not existed
1_2	Rls (Rear surround left) channel data is existed

The Rrs channel shows whether Rrs (Rear surround right) channel data is existed or not.

Table 12-34 – Rrs channel definition

Value	Description
0_2	Rrs (Rear surround right) channel data is not existed
1_2	Rrs (Rear surround right) channel data is existed

The L/R ch identifier shows whether L (Left)/R (Right) channel data is L/R signal (stereo) or M1 (mono) signal or Lo (Left output)/Ro (Right output) signal or Lt (Left total)/Rt (Right total) signal.
Note : Sink device shall decrease M1 (mono) signal at a -3dB level if Sink device outputs M1 (mono) signal in L (Left)/R (Right) channel.

Table 12-35 – L/R ch identifier definition

Value	Description
00_2	L (Left)/R (Right) channel data is L/R (stereo) signal
01_2	L (Left)/R (Right) channel data is M1 (mono) signal L (Left) channel and R (Right) channel data are the same
10_2	L (Left)/R (Right) channel data is Lo (Left output)/Ro (Right output) signal
11_2	L (Left)/R (Right) channel data is Lt (Left total)/Rt (Right total) signal

The C ch identifier shows whether C (Centre) channel data is C signal or M1 (mono) signal.

Table 12-36 – C ch identifier definition

Value	Description
0_2	C (Centre) channel data is C signal
1_2	C (Centre) channel data is M1 (mono) signal

The LS/RS ch identifier shows whether LS (Left Surround)/RS (Right Surround) channel data is LS/RS signal or S (Surround) signal.
Note : Sink device shall decrease S (Surround) signal at a -3dB level if Sink device outputs S (Surround) signal in LS (Left Surround)/RS (Right Surround) channel.

Table 12-37 – LS/RS ch identifier definition

Value	Description
0_2	LS (Left Surround)/RS (Right Surround) channel data is LS (Left Surround)/RS (Right Surround) signal
1_2	LS (Left Surround)/RS (Right Surround) channel data is S (Surround) signal LS (Left Surround) and RS (Right Surround) channel data are the same

12.4.5 Data for CCI

SUB LABEL $C0_{16}$ is for CCI.

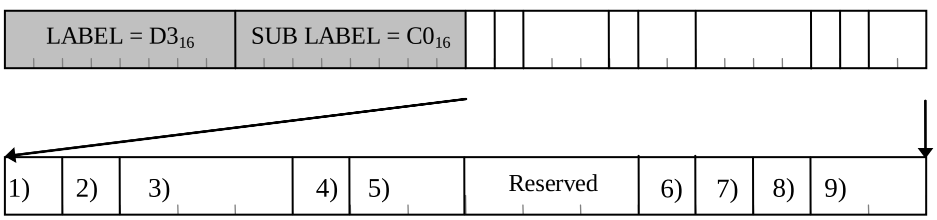

Figure 12-26 – Data for CCI

Table 12-38 – Data for CCI

Value	Bits
1) reserved	1
2) Retention_Move_Mode	1
3) Retention_State	3
4) EPN	1
5) CCI	2
6) Digital_Only_Token	1
7) Analog_Sunset_Token	1
8) Image_Constraint_Token	1
9) APS	2

Note: Ancillary data for CCI is for copy control information and is defined in [B5].

This data shall be transmitted at least once during 100mS period.

12.4.6 Example of Blu-ray Disc stream

Figure 12-27 – Basic data block of Blu-ray Disc stream illustrates a basic data block of Blu-ray Disc stream carried over the 1394 Bus in the case of eight channels.

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Data Channel 1 (Left channel)	
48_{16}	Data Channel 2 (Right channel)	
48_{16}	Data Channel 3 (low frequency effects channel)	
48_{16}	Data Channel 4 (Centre channel)	
48_{16}	Data Channel 5 (Left Surround channel)	
48_{16}	Data Channel 6 (Right Surround channel)	
48_{16}	Data Channel 7 (Rear surround left channel)	
48_{16}	Data Channel 8 (Rear surround right channel)	

Figure 12-27 – Basic data block of Blu-ray Disc

Figure 12-28 – Example of Blu-ray Disc Stream in the case of one channel (Mono) shows one channel case.

a) Example 1

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Mono	
48_{16}	Mono	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	

b) Example 2

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Mono	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	

c) Example 3

The following Channel Structure is permitted when only L and R are transmitted.

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Mono	
48_{16}	Mono	

Figure 12-28 – Examples of Blu-ray Disc Stream in the case of one channel

Figure 12-29 – Example of Blu-ray Disc Stream in the case of two channels (Left, Right) shows two channel cases.

a) Example 1

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Left	
48_{16}	Right	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	

b) Example 2

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Left	
48_{16}	Right	

Figure 12-29 – Example of Blu-ray Disc Stream in the case of two channels

Figure 12-30 – Example of Blu-ray Disc Stream in the case of three channels (3/0 : Left, Right, Centre) shows three channel cases.

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Left	
48_{16}	Right	
48_{16}	Set to 000000h	
48_{16}	Centre	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	

Figure 12-30 – Example of Blu-ray Disc Stream in the case of three channels (3/0)

Figure 12-31 – Example of Blu-ray Disc Stream in the case of three channels (2/1 : Left, Right, Surround) shows three channel cases.

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Left	
48_{16}	Right	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Surround	
48_{16}	Surround	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	

Figure 12-31 – Example of Blu-ray Disc Stream in the case of three channels (2/1)

Figure 12-32 – Example of Blu-ray Disc Stream in the case of four channels (2/2 : Left, Right, Left Surround, Right Surround) shows four channel cases.

$D3_{16}$	01_{16}	Ancillary Data
$D3_{16}$	$C0_{16}$	Ancillary Data
48_{16}	Left	
48_{16}	Right	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	
48_{16}	Left Surround	
48_{16}	Right Surround	
48_{16}	Set to 000000h	
48_{16}	Set to 000000h	

Figure 12-32 – Example of Blu-ray Disc Stream in the case of four channels (2/2)

12.5 Multi-bit Linear Audio (MBLA)

The compound data for Multi-bit Linear Audio consists of Multi-bit Linear Audio data, Common Ancillary and Multi-bit Linear Audio Specific Ancillary Data.

12.5.1 Structure of Sample Word for Audio transmission

There are two kind of channel structure in which this condition is defined:

1) Fixed Channels Structure

2) Variable Channels Structure

12.5.2 Fixed Channels Structure of Sample Word for Audio transmission

MBLA specifies thirty two channels. Thirty two channels are transmitted by four Isochronous channels.

There are eight Sample Word in one audio sample at each Isochronous channels.

Figure 12-33 - Basic data block of Fixed Channels Structure shows Channels included in each Group and channel order.

Group1

LABEL	Data Channel 1 (Front Left channel)
LABEL	Data Channel 2 (Front Right channel)
LABEL	Data Channel 3 (Low Frequency Effects-1 channel)
LABEL	Data Channel 4 (Front Centre channel)
LABEL	Data Channel 5 (Left Surround channel)
LABEL	Data Channel 6 (Right Surround channel)
LABEL	Data Channel 7 (Back Left channel)
LABEL	Data Channel 8 (Back Right channel)

Group2

LABEL	Data Channel 1 (Front Left centre channel)
LABEL	Data Channel 2 (Front Right centre channel)
LABEL	Data Channel 3 (Low Frequency Effects-2 channel)
LABEL	Data Channel 4 (Back Centre channel)
LABEL	Data Channel 5 (Side Left channel)
LABEL	Data Channel 6 (Side Right channel)
LABEL	Data Channel 7 (Top Front Left channel)
LABEL	Data Channel 8 (Top Front Right channel)

Group3

LABEL	Data Channel 1 (Front Left wide channel)
LABEL	Data Channel 2 (Front Right wide channel)
LABEL	Data Channel 3 (Top Front Centre channel)
LABEL	Data Channel 4 (Top Centre channel)
LABEL	Data Channel 5 (Top Back Left channel)
LABEL	Data Channel 6 (Top Back Right channel)
LABEL	Data Channel 7 (Top Side Left channel)
LABEL	Data Channel 8 (Top Side Right channel)

Group4

LABEL	Data Channel 1 (Top Back Centre channel)
LABEL	Data Channel 2 (Bottom Front Centre channel)
LABEL	Data Channel 3 (Bottom Front Left channel)
LABEL	Data Channel 4 (Bottom Front Right channel)
LABEL	Data Channel 5 (Left Surround direct channel)
LABEL	Data Channel 6 (Right Surround direct channel)
LABEL	Data Channel 7 (Top Left Surround channel)
LABEL	Data Channel 8 (Top Right Surround channel)

Figure 12-33 – Basic data block of Fixed Channels Structure

Channel Structure layout is fixed.

The transmitter shall set 000000_{16} on the audio data of MBLA data in the case of non existing channel.

The group does not need to transmit if all channels do not exist in the group.

12.5.3 Variable Channels Structure of Sample Word for Audio transmission

MBLA specifies thirty two channels. Thirty two channels are transmitted by one Isochronous channel.

Figure 12-34 - Basic data block of Variable channels Structure shows channel order.

LABEL	Data Channel 1 (Front Left channel)
LABEL	Data Channel 2 (Front Right channel)
LABEL	Data Channel 3 (Low Frequency Effects-1 channel)
LABEL	Data Channel 4 (Front Centre channel)
LABEL	Data Channel 5 (Left Surround channel)
LABEL	Data Channel 6 (Right Surround channel)
LABEL	Data Channel 7 (Back Left channel)
LABEL	Data Channel 8 (Back Right channel)

LABEL	Data Channel 9 (Front Left centre channel)
LABEL	Data Channel 10 (Front Right centre channel)
LABEL	Data Channel 11 (Low Frequency Effects-2 channel)
LABEL	Data Channel 12 (Back Centre channel)
LABEL	Data Channel 13 (Side Left channel)
LABEL	Data Channel 14 (Side Right channel)
LABEL	Data Channel 15 (Top Front Left channel)
LABEL	Data Channel 16 (Top Front Right channel)

LABEL	Data Channel 17 (Front Left wide channel)
LABEL	Data Channel 18 (Front Right wide channel)
LABEL	Data Channel 19 (Top Front Centre channel)
LABEL	Data Channel 20 (Top Centre channel)
LABEL	Data Channel 21 (Top Back Left channel)
LABEL	Data Channel 22 (Top Back Right channel)
LABEL	Data Channel 23 (Top Side Left channel)
LABEL	Data Channel 24 (Top Side Right channel)

LABEL	Data Channel 25 (Top Back Centre channel)
LABEL	Data Channel 26 (Bottom Front Centre channel)
LABEL	Data Channel 27 (Bottom Front Left channel)
LABEL	Data Channel 28 (Bottom Front Right channel)
LABEL	Data Channel 29 (Left Surround direct channel)
LABEL	Data Channel 30 (Right Surround direct channel)
LABEL	Data Channel 31 (Top Left Surround channel)
LABEL	Data Channel 32 (Top Right Surround channel)

Figure 12-34 – Basic data block of Variable Channels Structure

The non existing channel shall not transmit MBLA data.

Channel order is reduced.

12.5.4 MBLA data

MBLA data use the LABEL from 40_{16} to 42_{16} of MBLA.

12.5.5 MBLA Specific Ancillary Data

The clause specifies private header data that are carried by MBLA Specific Ancillary Data.

Table 12-39 – MBLA Specific Ancillary Data

LABEL	SUB LABEL	Description
$D4_{16}$	01_{16}	Data transmitted at every data block of Group 1 for Fixed Channels Structure
	02_{16}	Data transmitted at every data block of Group 2 for Fixed Channels Structure
	03_{16}	Data transmitted at every data block of Group 3 for Fixed Channels Structure
	04_{16}	Data transmitted at every data block of Group 4 for Fixed Channels Structure
	05_{16}	Data transmitted at every data block for Variable Channels Structure
	06_{16}	Data transmitted at Extension Channel Bit Order 1 for Variable Channels Structure
	07_{16}	Data transmitted at Extension Channel Bit Order 2 for Variable Channels Structure
	$C0_{16}$	CCI

The transmission device shall execute stream change method if Ancillary Data is changed except when SUB LABEL is $C0_{16}$.

12.5.6 Data transmitted at every data block of Group 1 for Fixed Channels Structure

This Ancillary Data is transmitted at every data block of Group 1 for Fixed Channels Structure.

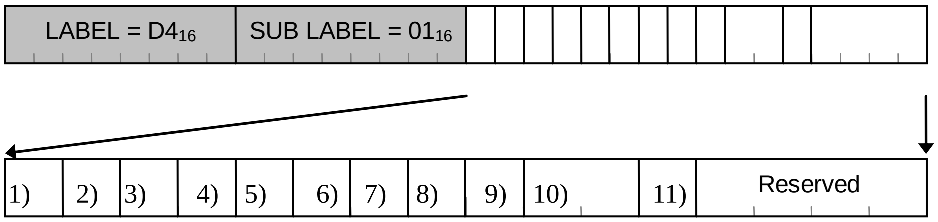

Figure 12-35 – Data transmitted at every data block of Group 1 for Fixed Channels Structure

Table 12-40 – Data transmitted at every data block of Group 1 for Fixed Channels Structure

Data	Bits	Description
1) Emphasis Flag	1	Emphasis on or off
2) FL channel	1	FL (Front Left) channel data exist or not exist
3) FR channel	1	FR (Front Right) channel data exist or not exist
4) LFE1 channel	1	LFE1 (Low Frequency Effects-1) channel data exist or not exist
5) FC channel	1	FC (Front Centre) channel data exist or not exist
6) LS channel	1	LS (Left Surround) channel data exist or not exist
7) RS channel	1	RS (Right Surround) channel data exist or not exist
8) BL channel	1	BL (Back Left) channel data exist or not exist
9) BR channel	1	BR (Back Right) channel data exist or not exist
10) FL/FR ch identifier	2	FL (Front Left)/FR (Front Right) channel identifier defined
11) FC ch identifier	1	FC (Front Centre) channel identifier defined

The Emphasis Flag shows whether de-emphasis is required for the sink device or not.

Table 12-41 – Emphasis Flag definition

Value	Description
0_2	de-emphasis is not required
1_2	de-emphasis is required

The FL channel shows whether FL (Front Left) channel data is existed or not.

Table 12-42 – FL channel definition

Value	Description
0_2	FL (Front Left) channel data is not existed
1_2	FL (Front Left) channel data is existed

The FR channel shows whether FR (Front Right) channel data is existed or not.

Table 12-43 – FR channel definition

Value	Description
0_2	FR (Front Right) channel data is not existed
1_2	FR (Front Right) channel data is existed

The LFE1 channel shows whether LFE1 (Low Frequency Effects-1) channel data is existed or not.

Table 12-44 – LFE1 channel definition

Value	Description
0_2	LFE1 (Low Frequency Effects-1) channel data is not existed
1_2	LFE1 (Low Frequency Effects-1) channel data is existed

The FC channel shows whether FC (Front Centre) channel data is existed or not.

Table 12-45 – FC channel definition

Value	Description
0_2	FC (Front Centre) channel data is not existed
1_2	FC (Front Centre) channel data is existed

The LS channel shows whether LS (Left Surround) channel data is existed or not.

Table 12-46 – LS channel definition

Value	Description
0_2	LS (Left Surround) channel data is not existed
1_2	LS (Left Surround) channel data is existed

The RS channel shows whether RS (Right Surround) channel data is existed or not.

Table 12-47 – RS channel definition

Value	Description
0_2	RS (Right Surround) channel data is not existed
1_2	RS (Right Surround) channel data is existed

The BL channel shows whether BL (Back Left) channel data is existed or not.

Table 12-48 – BL channel definition

Value	Description
0_2	BL (Back Left) channel data is not existed
1_2	BL (Back Left) channel data is existed

The BR channel shows whether BR (Back Right) channel data is existed or not

Table 12-49 – BR channel definition

Value	Description
0_2	BR (Back Right) channel data is not existed
1_2	BR (Back Right) channel data is existed

The FL/FR ch identifier shows whether FL (Front Left)/FR (Front Right) channel data is FL/FR signal (stereo) or M1 (mono) signal or Lo (Left output)/Ro (Right output) signal or Lt (Left total)/Rt (Right total) signal.

Note : Sink device shall decrease M1 (mono) signal at a -3dB level if Sink device outputs M1 (mono) signal in L (Left)/R (Right) channel.

Table 12-50 – FL/FR ch identifier definition

Value	Description
00_2	FL (Front Left)/FR (Front Right) channel data is L/R (stereo) signal
01_2	FL (Front Left)/FR (Front Right) channel data is M1 (mono) signal FL (Front Left) channel and FR (Front Right) channel data are the same
10_2	FL (Front Left)/FR (Front Right) channel data is Lo (Left output)/Ro (Right output) signal
11_2	FL (Front Left)/FR (Front Right) channel data is Lt (Left total)/Rt (Right total) signal

The FC ch identifier shows whether FC (Front Centre) channel data is FC signal or M1 (mono) signal.

1394 TRADE ASSOCIATION

Table 12-51 – FC ch identifier definition

Value	Description
0_2	FC (Front Centre) channel data is FC signal
1_2	FC (Front Centre) channel data is M1 (mono) signal

12.5.7 Data transmitted at every data block of Group 2 for Fixed Channels Structure

This Ancillary Data is transmitted at every data block of Group 2 for Fixed Channels Structure.

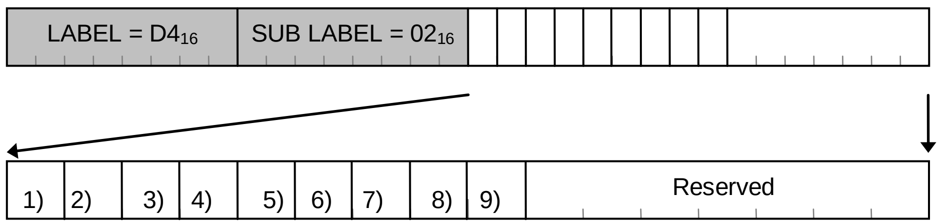

Figure 12-36 – Data transmitted at every data of Group 2 for Fixed Channels Structure

Table 12-52 – Data transmitted at every data of Group 2 for Fixed Channels Structure

Data	Bits	Description
1) Emphasis Flag	1	Emphasis on or off
2) FLc channel	1	FLc (Front Left centre) channel data exist or not exist
3) FRc channel	1	FRc (Front Right centre) channel data exist or not exist
4) LFE2 channel	1	LFE2 (Low Frequency Effects-2) channel data exist or not exist
5) BC channel	1	BC (Back Centre) channel data exist or not exist
6) SiL channel	1	SiL (Side Left) channel data exist or not exist
7) SiR channel	1	SiR (Side Right) channel data exist or not exist
8) TpFL channel	1	TpFL (Top Front Left) channel data exist or not exist
9) TpFR channel	1	TpFR (Top Front Right) channel data exist or not exist

The Emphasis Flag shows whether de-emphasis is required for the sink device or not.

Table 12-53 – Emphasis Flag definition

Value	Description
0_2	de-emphasis is not required
1_2	de-emphasis is required

The FLc channel shows whether FLc (Front Left centre) channel data is existed or not.

Table 12-54 – FLc channel definition

Value	Description
0_2	FLc (Front Left centre) channel data is not existed
1_2	FLc (Front Left centre) channel data is existed

The FRc channel shows whether FRc (Front Right centre) channel data is existed or not.

Table 12-55 – FRc channel definition

Value	Description
0_2	FRc (Front Right centre) channel data is not existed
1_2	FRc (Front Right centre) channel data is existed

The LFE2 channel shows whether LFE2 (Low Frequency Effects-2) channel data is existed or not.

Table 12-56 – LFE2 channel definition

Value	Description
0_2	LFE2 (Low Frequency Effects-2) channel data is not existed
1_2	LFE2 (Low Frequency Effects-2) channel data is existed

The BC channel shows whether BC (Back Centre) channel data is existed or not.

Table 12-57 – BC channel definition

Value	Description
0_2	BC (Back Centre) channel data is not existed
1_2	BC (Back Centre) channel data is existed

The SiL channel shows whether SiL (Side Left) channel data is existed or not.

Table 12-58 – SiL channel definition

Value	Description
0_2	SiL (Side Left) channel data is not existed
1_2	SiL (Side Left) channel data is existed

The SiR channel shows whether SiR (Side Right) channel data is existed or not.

Table 12-59 – SiR channel definition

Value	Description
0_2	SiR (Side Right) channel data is not existed
1_2	SiR (Side Right) channel data is existed

The TpFL channel shows whether TpFL (Top Front Left) channel data is existed or not.

Table 12-60 – TpFL channel definition

Value	Description
0_2	TpFL (Top Front Left) channel data is not existed
1_2	TpFL (Top Front Left) channel data is existed

The TpFR channel shows whether TpFR (Top Front Right) channel data is existed or not.

Table 12-61 – TpFR channel definition

Value	Description
0_2	TpFR (Top Front Right) channel data is not existed
1_2	TpFR (Top Front Right) channel data is existed

12.5.8 Data transmitted at every data block of Group 3 for Fixed Channels Structure

This Ancillary Data is transmitted at every data block of Group 3 for Fixed Channels Structure.

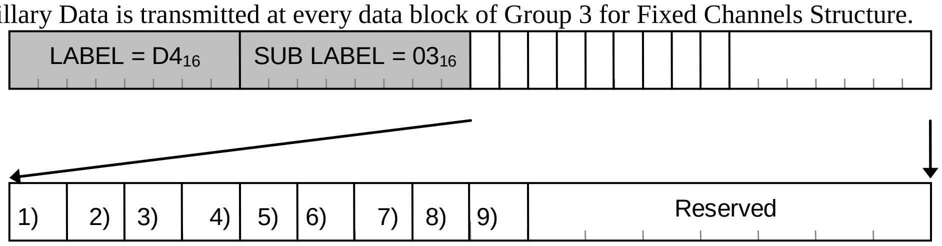

Figure 12-37 – Data transmitted at every data block of Group 3 for Fixed Channels Structure

Table 12-62 – Data transmitted at every data block of Group 3 for Fixed Channels Structure

Data	Bits	Description
1) Emphasis Flag	1	Emphasis on or off
2) FLw channel	1	FLw (Front Left wide) channel data exist or not exist
3) FRw channel	1	FRw (Front Right wide) channel data exist or not exist
4) TpFC channel	1	TpFC (Top Front Centre) channel data exist or not exist
5) TpC channel	1	TpC (Top Centre) channel data exist or not exist
6) TpBL channel	1	TpBL (Top Back Left) channel data exist or not exist
7) TpBR channel	1	TpBR (Top Back Right) channel data exist or not exist
8) TpSiL channel	1	TpSiL (Top Side Left) channel data exist or not exist
9) TpSiR channel	1	TpSiR (Top Side Right) channel data exist or not exist

The Emphasis Flag shows whether de-emphasis is required for the sink device or not.

Table 12-63 – Emphasis Flag definition

Value	Description
0_2	de-emphasis is not required
1_2	de-emphasis is required

The FLw channel shows whether FLw (Front Left wide) channel data is existed or not.

Table 12-64 – FLw channel definition

Value	Description
0_2	FLw (Front Left wide) channel data is not existed
1_2	FLw (Front Left wide) channel data is existed

The FRw channel shows whether FRw (Front Right wide) channel data is existed or not.

Table 12-65 – FRw channel definition

Value	Description
0_2	FRw (Front Right wide) channel data is not existed
1_2	FRw (Front Right wide) channel data is existed

The TpFC channel shows whether TpFC (Top Front Centre) channel data is existed or not.

Table 12-66 – TpFC channel definition

Value	Description
0_2	TpFC (Top Front Centre) channel data is not existed
1_2	TpFC (Top Front Centre) channel data is existed

The TpC channel shows whether TpC (Top Centre) channel data is existed or not.

Table 12-67 – TpC channel definition

Value	Description
0_2	TpC (Top Centre) channel data is not existed
1_2	TpC (Top Centre) channel data is existed

The TpBL channel shows whether TpBL (Top Back Left) channel data is existed or not.

Table 12-68 – TpBL channel definition

Value	Description
0_2	TpBL (Top Back Left) channel data is not existed
1_2	TpBL (Top Back Left) channel data is existed

The TpBR channel shows whether TpBR (Top Back Right) channel data is existed or not.

Table 12-69 – TpBR channel definition

Value	Description
0_2	TpBR (Top Back Right) channel data is not existed
1_2	TpBR (Top Back Right) channel data is existed

The TpSiL channel shows whether TpSiL (Top Side Left) channel data is existed or not.

Table 12-70 – TpSiL channel definition

Value	Description
0_2	TpSiL (Top Side Left) channel data is not existed
1_2	TpSiL (Top Side Left) channel data is existed

The TpSiR channel shows whether TpSiR (Top Side Right) channel data is existed or not.

Table 12-71 – TpSiR channel definition

Value	Description
0_2	TpSiR (Top Side Right) channel data is not existed
1_2	TpSiR (Top Side Right) channel data is existed

12.5.9 Data transmitted at every data block of Group 4 for Fixed Channels Structure

This Ancillary Data is transmitted at every data block of Group 4 for Fixed Channels Structure.

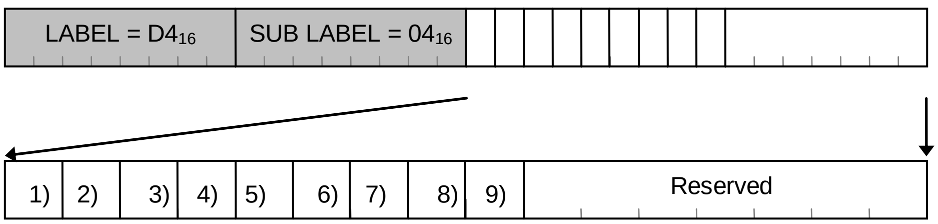

Figure 12-38 – Data transmitted at every data block of Group 4 for Fixed Channels Structure

Table 12-72 – Data transmitted at every data block of Group 4 for Fixed Channels Structure

Data	Bits	Description
1) Emphasis Flag	1	Emphasis on or off
2) TpBC channel	1	TpBC (Top Back Centre) channel data exist or not exist
3) BtFC channel	1	BtFC (Bottom Front Centre) channel data exist or not exist
4) BtFL channel	1	BtFL (Bottom Front Left) channel data exist or not exist
5) BtFR channel	1	BtFR (Bottom Front Right) channel data exist or not exist
6) LSd channel	1	LSd (Left Surround direct) channel data exist or not exist
7) RSd channel	1	RSd (Right Surround direct) channel data exist or not exist
8) TpLS channel	1	TpLS (Top Left Surround) channel data exist or not exist
9) TpRS channel	1	TpRS (Top Right Surround) channel data exist or not exist

The Emphasis Flag shows whether de-emphasis is required for the sink device or not.

Table 12-73 – Emphasis Flag definition

Value	Description
0_2	de-emphasis is not required
1_2	de-emphasis is required

The TpBC channel shows whether TpBC (Top Back Centre) channel data is existed or not.

Table 12-74 – TpBC channel definition

Value	Description
0_2	TpBC (Top Back Centre) channel data is not existed
1_2	TpBC (Top Back Centre) channel data is existed

The BtFC channel shows whether BtFC (Bottom Front Centre) channel data is existed or not.

Table 12-75 – BtFC channel definition

Value	Description
0_2	BtFC (Bottom Front Centre) channel data is not existed
1_2	BtFC (Bottom Front Centre) channel data is existed

The BtFL channel shows whether BtFL (Bottom Front Left) channel data is existed or not.

Table 12-76 – BtFL channel definition

Value	Description
0_2	BtFL (Bottom Front Left) channel data is not existed
1_2	BtFL (Bottom Front Left) channel data is existed

The BtFR channel shows whether BtFR (Bottom Front Right) channel data is existed or not.

Table 12-77 – BtFR channel definition

Value	Description
0_2	BtFR (Bottom Front Right) channel data is not existed
1_2	BtFR (Bottom Front Right) channel data is existed

The LSd channel shows whether LSd (Left Surround direct) channel data is existed or not.

Table 12-78 – LSd channel definition

Value	Description
0_2	LSd (Left Surround direct) channel data is not existed
1_2	LSd (Left Surround direct) channel data is existed

The RSd channel shows whether RSd (Right Surround direct) channel data is existed or not.

Table 12-79 – RSd channel definition

Value	Description
0_2	RSd (Right Surround direct) channel data is not existed
1_2	RSd (Right Surround direct) channel data is existed

The TpLS channel shows whether TpLS (Top Left Surround) channel data is existed or not.

Table 12-80 – TpLS channel definition

Value	Description
0_2	TpLS (Top Left Surround) channel data is not existed
1_2	TpLS (Top Left Surround) channel data is existed

The TpRS channel shows whether TpRS (Top Right Surround) channel data is existed or not.

Table 12-81 – TpRS channel definition

Value	Description
0_2	TpRS (Top Right Surround) channel data is not existed
1_2	TpRS (Top Right Surround) channel data is existed

12.5.10 Data transmitted at every data block for Variable Channels Structure

This Ancillary Data is transmitted at every data block for Variable Channels Structure.

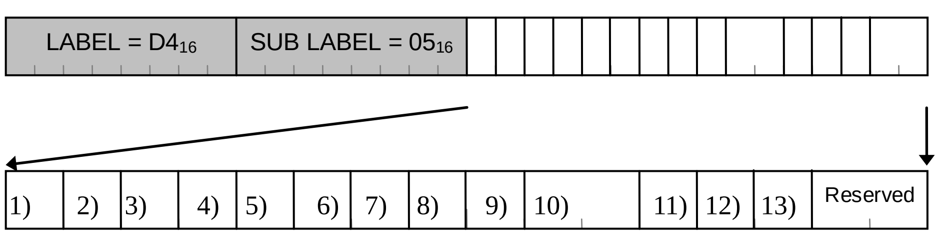

Figure 12-39 – Data transmitted at every data block for Variable Channels Structure

Table 12-82 – Data transmitted at every data block for Variable Channels Structure

Data	Bits	Description
1) Emphasis Flag	1	Emphasis on or off
2) FL channel	1	FL (Front Left) channel data exist or not exist
3) FR channel	1	FR (Front Right) channel data exist or not exist
4) LFE1 channel	1	LFE1 (Low Frequency Effects-1) channel data exist or not exist
5) FC channel	1	FC (Front Centre) channel data exist or not exist
6) LS channel	1	LS (Left Surround) channel data exist or not exist
7) RS channel	1	RS (Right Surround) channel data exist or not exist
8) BL channel	1	BL (Back Left) channel data exist or not exist
9) BR channel	1	BR (Back Right) channel data exist or not exist
10) FL/FR ch identifier	2	FL (Front Left)/FR (Front Right) channel identifier defined
11) FC ch identifier	1	FC (Front Centre) channel identifier defined
12) Extension Ch Flag 1	1	Extension Channel Bit Order 1 exist or not exist
13) Extension Ch Flag 2	1	Extension Channel Bit Order 2 exist or not exist

The Emphasis Flag shows whether de-emphasis is required for the sink device or not.

Table 12-83 – Emphasis Flag definition

Value	Description
0_2	de-emphasis is not required
1_2	de-emphasis is required

The FL channel shows whether FL (Front Left) channel data is existed or not.

Table 12-84 – FL channel definition

Value	Description
0_2	FL (Front Left) channel data is not existed
1_2	FL (Front Left) channel data is existed

The FR channel shows whether FR (Front Right) channel data is existed or not.

Table 12-85 – FR channel definition

Value	Description
0_2	FR (Front Right) channel data is not existed
1_2	FR (Front Right) channel data is existed

The LFE1 channel shows whether LFE1 (Low Frequency Effects-1) channel data is existed or not.

Table 12-86 – LFE1 channel definition

Value	Description
0_2	LFE1 (Low Frequency Effects-1) channel data is not existed
1_2	LFE1 (Low Frequency Effects-1) channel data is existed

The FC channel shows whether FC (Front Centre) channel data is existed or not.

Table 12-87 – FC channel definition

Value	Description
0_2	FC (Front Centre) channel data is not existed
1_2	FC (Front Centre) channel data is existed

The LS channel shows whether LS (Left Surround) channel data is existed or not.

Table 12-88 – LS channel definition

Value	Description
0_2	LS (Left Surround) channel data is not existed
1_2	LS (Left Surround) channel data is existed

The RS channel shows whether RS (Right Surround) channel data is existed or not.

Table 12-89 – RS channel definition

Value	Description
0_2	RS (Right Surround) channel data is not existed
1_2	RS (Right Surround) channel data is existed

The BL channel shows whether BL (Back Left) channel data is existed or not.

Table 12-90 – BL channel definition

Value	Description
0_2	BL (Back Left) channel data is not existed
1_2	BL (Back Left) channel data is existed

The BR channel shows whether BR (Back Right) channel data is existed or not.

Table 12-91 – BR channel definition

Value	Description
0_2	BR (Back Right) channel data is not existed
1_2	BR (Back Right) channel data is existed

The FL/FR ch identifier shows whether FL (Front Left)/FR (Front Right) channel data is FL/FR signal (stereo) or M1 (mono) signal or Lo (Left output)/Ro (Right output) signal or Lt (Left total)/Rt (Right total) signal.

Note : Sink device shall decrease M1 (mono) signal at a -3dB level if Sink device outputs M1 (mono) signal in L (Left)/R (Right) channel.

Table 12-92 – FL/FR ch identifier definition

Value	Description
00_2	FL (Front Left)/FR (Front Right) channel data is L/R (stereo) signal
01_2	FL (Front Left)/FR (Front Right) channel data is M1 (mono) signal FL (Front Left) channel and FR (Front Right) channel data are the same
10_2	FL (Front Left)/FR (Front Right) channel data is Lo (Left output)/Ro (Right output) signal
11_2	FL (Front Left)/FR (Front Right) channel data is Lt (Left total)/Rt (Right total) signal

The FC ch identifier shows whether FC (Front Centre) channel data is FC signal or M1 (mono) signal.

Table 12-93 – FC ch identifier definition

Value	Description
0_2	FC (Front Centre) channel data is FC signal
1_2	FC (Front Centre) channel data is M1 (mono) signal

The Extension Ch Flag 1 shows whether Ancillary data of Data transmitted at Extension Channel Bit Order 1 for Variable Channels Structure is existed or not.

Table 12-94 – Extension Ch Flag 1 definition

Value	Description
0_2	Ancillary data of Data transmitted at Extension Channel Bit Order 1 for Variable Channels Structure is not existed
1_2	Ancillary data of Data transmitted at Extension Channel Bit Order 1 for Variable Channels Structure is existed

The Extension Ch Flag 2 shows whether Ancillary date of Data transmitted at Extension Channel Bit Order 2 for Variable Channels Structure is existed or not.

Table 12-95 – Extension Ch Flag 2 definition

Value	Description
0_2	Ancillary date of Data transmitted at Extension Channel Bit Order 2 for Variable Channels Structure is not existed
1_2	Ancillary date of Data transmitted at Extension Channel Bit Order 2 for Variable Channels Structure is existed

12.5.11 Data transmitted at Extension Channel Bit Order 1 for Variable Channels Structure

This Ancillary Data is transmitted at Extension Channel Bit Order 1 for Variable Channels Structure.

This data shall be transmitted as Second Ancillary Data at least once per less than 100ms when this data exist.

The transmitter should output this Ancillary data as soon as possible to clarify the channel assignment after contents of the stream were changed or the stream output started.

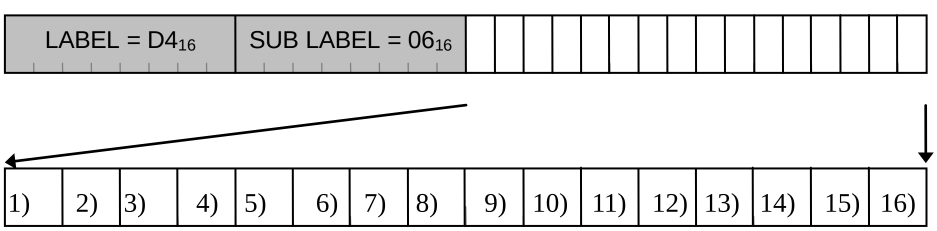

Figure 12-40 – Data transmitted at Extension Channel Bit Order 1 for Variable Channels Structure

Table 12-96 – Data transmitted at Extension Channel Bit Order 1 for Variable Channels Structure

Data	Bits	Description
1) FLc channel	1	FLc (Front Left centre) channel data exist or not exist
2) FRc channel	1	FRc (Front Right centre) channel data exist or not exist
3) LFE2 channel	1	LFE2 (Low Frequency Effects-2) channel data exist or not exist
4) BC channel	1	BC (Back Centre) channel data exist or not exist
5) SiL channel	1	SiL (Side Left) channel data exist or not exist
6) SiR channel	1	SiR (Side Right) channel data exist or not exist
7) TpFL channel	1	TpFL (Top Front Left) channel data exist or not exist
8) TpFR channel	1	TpFR (Top Front Right) channel data exist or not exist
9) FLw channel	1	FLw (Front Left wide) channel data exist or not exist
10) FRw channel	1	FRw (Front Right wide) channel data exist or not exist
11) TpFC channel	1	TpFC (Top Front Centre) channel data exist or not exist
12) TpC channel	1	TpC (Top Centre) channel data exist or not exist
13) TpBL channel	1	TpBL (Top Back Left) channel data exist or not exist
14) TpBR channel	1	TpBR (Top Back Right) channel data exist or not exist
15) TpSiL channel	1	TpSiL (Top Side Left) channel data exist or not exist
16) TpSiR channel	1	TpSiR (Top Side Right) channel data exist or not exist

The FLc channel shows whether FLc (Front Left centre) channel data is existed or not.

Table 12-97 – FLc channel definition

Value	Description
0_2	FLc (Front Left centre) channel data is not existed
1_2	FLc (Front Left centre) channel data is existed

The FRc channel shows whether FRc (Front Right centre) channel data is existed or not.

Table 12-98 – FRc channel definition

Value	Description
0_2	FRc (Front Right centre) channel data is not existed
1_2	FRc (Front Right centre) channel data is existed

The LFE2 channel shows whether LFE2 (Low Frequency Effects-2) channel data is existed or not.

Table 12-99 – LFE2 channel definition

Value	Description
0_2	LFE2 (Low Frequency Effects-2) channel data is not existed
1_2	LFE2 (Low Frequency Effects-2) channel data is existed

The BC channel shows whether BC (Back Centre) channel data is existed or not.

Table 12-100 – BC channel definition

Value	Description
0_2	BC (Back Centre) channel data is not existed
1_2	BC (Back Centre) channel data is existed

The SiL channel shows whether SiL (Side Left) channel data is existed or not.

Table 12-101 – SiL channel definition

Value	Description
0_2	SiL (Side Left) channel data is not existed
1_2	SiL (Side Left) channel data is existed

The SiR channel shows whether SiR (Side Right) channel data is existed or not.

Table 12-102 – SiR channel definition

Value	Description
0_2	SiR (Side Right) channel data is not existed
1_2	SiR (Side Right) channel data is existed

The TpFL channel shows whether TpFL (Top Front Left) channel data is existed or not.

Table 12-103 – TpFL channel definition

Value	Description
0_2	TpFL (Top Front Left) channel data is not existed
1_2	TpFL (Top Front Left) channel data is existed

The TpFR channel shows whether TpFR (Top Front Right) channel data is existed or not.

Table 12-104 – TpFR channel definition

Value	Description
0_2	TpFR (Top Front Right) channel data is not existed
1_2	TpFR (Top Front Right) channel data is existed

The FLw channel shows whether FLw (Front Left wide) channel data is existed or not.

Table 12-105 – FLw channel definition

Value	Description
0_2	FLw (Front Left wide) channel data is not existed
1_2	FLw (Front Left wide) channel data is existed

The FRw channel shows whether FRw (Front Right wide) channel data is existed or not.

Table 12-106 – FRw channel definition

Value	Description
0_2	FRw (Front Right wide) channel data is not existed
1_2	FRw (Front Right wide) channel data is existed

The TpFC channel shows whether TpFC (Top Front Centre) channel data is existed or not.

Table 12-107 – TpFC channel definition

Value	Description
0_2	TpFC (Top Front Centre) channel data is not existed
1_2	TpFC (Top Front Centre) channel data is existed

The TpC channel shows whether TpC (Top Centre) channel data is existed or not.

Table 12-108 – TpC channel definition

Value	Description
0_2	TpC (Top Centre) channel data is not existed
1_2	TpC (Top Centre) channel data is existed

The TpBL channel shows whether TpBL (Top Back Left) channel data is existed or not.

Table 12-109 – TpBL channel definition

Value	Description
0_2	TpBL (Top Back Left) channel data is not existed
1_2	TpBL (Top Back Left) channel data is existed

The TpBR channel shows whether TpBR (Top Back Right) channel data is existed or not.

Table 12-110 – TpBR channel definition

Value	Description
0_2	TpBR (Top Back Right) channel data is not existed
1_2	TpBR (Top Back Right) channel data is existed

The TpSiL channel shows whether TpSiL (Top Side Left) channel data is existed or not.

Table 12-111 – TpSiL channel definition

Value	Description
0_2	TpSiL (Top Side Left) channel data is not existed
1_2	TpSiL (Top Side Left) channel data is existed

The TpSiR channel shows whether TpSiR (Top Side Right) channel data is existed or not.

Table 12-112 – TpSiR channel definition

Value	Description
0_2	TpSiR (Top Side Right) channel data is not existed
1_2	TpSiR (Top Side Right) channel data is existed

12.5.12 Data transmitted at Extension Channel Bit Order 2 for Variable Channels Structure

This Ancillary Data is transmitted at Extension Channel Bit Order 2 for Variable Channels Structure.

This data shall be transmitted as Second Ancillary Data at least once per less than 100ms when this data exist.

The transmitter should output this Ancillary data as soon as possible to clarify the channel assignment after contents of the stream were changed or the stream output started.

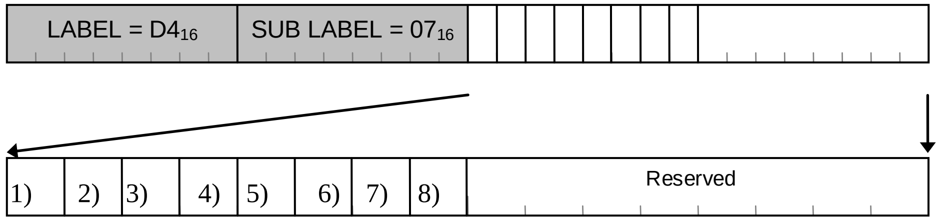

Figure 12-41 – Data transmitted at Extension Channel Bit Order 2 for Variable Channels Structure

Table 12-113 – Data transmitted at Extension Channel Bit Order 2 for Variable Channels Structure

Data	Bits	Description
1) TpBC channel	1	TpBC (Top Back Centre) channel data exist or not exist
2) BtFC channel	1	BtFC (Bottom Front Centre) channel data exist or not exist
3) BtFL channel	1	BtFL (Bottom Front Left) channel data exist or not exist
4) BtFR channel	1	BtFR (Bottom Front Right) channel data exist or not exist
5) LSd channel	1	LSd (Left Surround direct) channel data exist or not exist
6) RSd channel	1	RSd (Right Surround direct) channel data exist or not exist
7) TpLS channel	1	TpLS (Top Left Surround) channel data exist or not exist
8) TpRS channel	1	TpRS (Top Right Surround) channel data exist or not exist

The TpBC channel shows whether TpBC (Top Back Centre) channel data is existed or not.

Table 12-114 – TpBC channel definition

Value	Description
0_2	TpBC (Top Back Centre) channel data is not existed
1_2	TpBC (Top Back Centre) channel data is existed

The BtFC channel shows whether BtFC (Bottom Front Centre) channel data is existed or not.

Table 12-115 – BtFC channel definition

Value	Description
0_2	BtFC (Bottom Front Centre) channel data is not existed
1_2	BtFC (Bottom Front Centre) channel data is existed

The BtFL channel shows whether BtFL (Bottom Front Left) channel data is existed or not.

Table 12-116 – BtFL channel definition

Value	Description
0_2	BtFL (Bottom Front Left) channel data is not existed
1_2	BtFL (Bottom Front Left) channel data is existed

The BtFR channel shows whether BtFR (Bottom Front Right) channel data is existed or not.

Table 12-117 – BtFR channel definition

Value	Description
0_2	BtFR (Bottom Front Right) channel data is not existed
1_2	BtFR (Bottom Front Right) channel data is existed

The LSd channel shows whether LSd (Left Surround direct) channel data is existed or not.

Table 12-118 – LSd channel definition

Value	Description
0_2	LSd (Left Surround direct) channel data is not existed
1_2	LSd (Left Surround direct) channel data is existed

The RSd channel shows whether RSd (Right Surround direct) channel data is existed or not.

Table 12-119 – RSd channel definition

Value	Description
0_2	RSd (Right Surround direct) channel data is not existed
1_2	RSd (Right Surround direct) channel data is existed

The TpLS channel shows whether TpLS (Top Left Surround) channel data is existed or not.

Table 12-120 – TpLS channel definition

Value	Description
0_2	TpLS (Top Left Surround) channel data is not existed
1_2	TpLS (Top Left Surround) channel data is existed

The TpRS channel shows whether TpRS (Top Right Surround) channel data is existed or not.

Table 12-121 – TpRS channel definition

Value	Description
0_2	TpRS (Top Right Surround) channel data is not existed
1_2	TpRS (Top Right Surround) channel data is existed

12.5.13 Data for CCI

SUB LABEL $C0_{16}$ is for CCI.

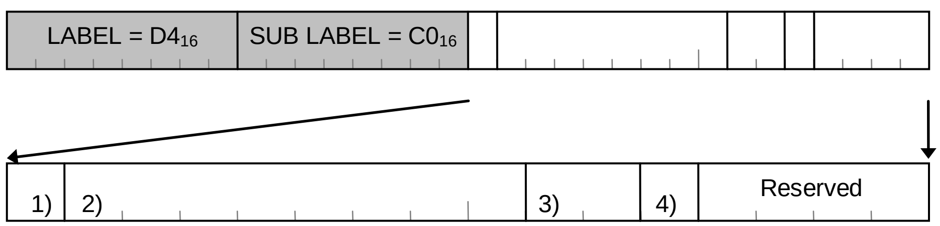

Figure 12-42 – Ancillary Data for CCI

Table 12-122 – Data transmitted at every data block

Data	Bits
1) CP-bit	1
2) Category code	8
6) CGMS-A	2
5) CGMS-A validity	1

Note: Ancillary data for CCI contains the same meaning specified in IEC60958-3.

Note: Each value is based on the following condition in IEC60958-3.

Consumer use of channel status block

Audio sample word represents linear PCM samples

Channel status mode is Mode 0

This data shall be transmitted at least once during 100mS period.

12.5.14 Example of MBLA stream for Fixed Channels Structure

Figure 12-43 – Example of MBLA stream for Fixed Channels Structure in the case of one channel (Mono) shows one channel case.

a) Example 1

$D4_{16}$	01_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Mono	
42_{16}	Mono	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	

b) Example 2

$D4_{16}$	01_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Mono	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	

c) Example 3

The following Channel Structure is permitted when only FL and FR are transmitted.

$D4_{16}$	01_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Mono	
42_{16}	Mono	

Figure 12-43 – Example of MBLA stream for Fixed Channels Structure in the case of one channel

Figure 12-44 – Example of MBLA stream for Fixed Channels Structure in the case of two channels (Front Left, Front Right) shows two channel cases.

a) Example 1

$D4_{16}$	01_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Front Left	
42_{16}	Front Right	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	

b) Example 2

The following Channel Structure is permitted when only FL and FR are transmitted.

$D4_{16}$	01_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Front Left	
42_{16}	Front Right	

Figure 12-44 – Example of MBLA stream for Fixed Channels Structure in the case of two channels

Figure 12-45 – Example of MBLA stream for Fixed Channels Structure in the case of three channels (3/0 : Front Left, Front Right, Front Centre) shows three channel cases.

$D4_{16}$	01_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Front Left	
42_{16}	Front Right	
42_{16}	Set to 000000_{16}	
42_{16}	Front Centre	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	

Figure 12-45 – Example of MBLA stream for Fixed Channels Structure in the case of three channels (3/0)

Figure 12-46 – Example of MBLA stream for Fixed Channels Structure in the case of four channels (2/2 : Front Left, Front Right, Left Surround, Right Surround) shows four channel cases.

$D4_{16}$	01_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Front Left	
42_{16}	Front Right	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	
42_{16}	Left Surround	
42_{16}	Right Surround	
42_{16}	Set to 000000_{16}	
42_{16}	Set to 000000_{16}	

Figure 12-46 – Example of MBLA stream for Fixed Channels Structure in the case of four channels (2/2)

12.5.15 Example of MBLA stream for Variable Channels Structure

Figure 12-47 – Example of MBLA stream for Variable Channels Structure in the case of one channel (Mono) shows one channel case.

a) Example 1

$D4_{16}$	05_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Mono	
42_{16}	Mono	

b) Example 2

$D4_{16}$	05_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Mono	
CF_{16}	CF_{16}	

Figure 12-47 – Example of MBLA stream for Variable Channels Structure in the case of one channel

Figure 12-48 – Example of MBLA stream for Variable Channels Structure in the case of two channels (Front Left, Front Right) shows two channel cases.

$D4_{16}$	05_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Front Left	
42_{16}	Front Right	

Figure 12-48 – Example of MBLA stream for Variable Channels Structure in the case of two channels

Figure 12-49 – Example of MBLA stream for Variable Channels Structure in the case of three channels (3/0 : Front Left, Front Right, Front Centre) shows three channel cases.

$D4_{16}$	05_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Front Left	
42_{16}	Front Right	
42_{16}	Front Centre	
CF_{16}	CF_{16}	

Figure 12-49 – Example of MBLA stream for Variable Channels Structure in the case of three channels (3/0)

Figure 12-50 – Example of MBLA stream for Variable Channels Structure in the case of four channels (2/2 : Front Left, Front Right, Left Surround, Right Surround) shows four channel cases.

$D4_{16}$	05_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Front Left	
42_{16}	Front Right	
42_{16}	Left Surround	
42_{16}	Right Surround	

Figure 12-50 – Example of MBLA stream for Variable Channels Structure in the case of four channels (2/2)

Figure 12-51 – Example of MBLA stream for Variable Channels Structure in the case of seven channels (Front Left, Front Right, Front Centre, Left Surround, Right Surround, Front Left wide, Front Right wide) shows seven channel cases.

$D4_{16}$	01_{16}	Ancillary Data
$D4_{16}$	$C0_{16}$	Ancillary Data
42_{16}	Front Left	
42_{16}	Front Right	
42_{16}	Front Centre	
42_{16}	Left Surround	
42_{16}	Right Surround	
42_{16}	Front Left wide	
42_{16}	Front Right wide	
CF_{16}	CF_{16}	

Figure 12-51 – Example of MBLA stream for Variable Channels Structure in the case of seven channels

Annexes

Annex A: Bibliography (informative)

[B1] Super Audio CD System Description Version 1.2

[B2] DVD Specifications for Read-Only Disc Part 4, Audio Specifications Version 1.0 March 1999

[B3] DVD Specifications for Read-Only Disc Part 4, Audio Specifications Version-up Information (from 1.0 to 1.1) May 1999

[B4] DVD Specifications for Read-Only Disc Part 4, Audio Specifications Version-up Information (from 1.1 to 1.2) May 2000

[B5] System Description Blu-ray Disc Read-Only Format Part 3 Audio Visual Basic SpecificationsVersion 2.5

[B6] SMPTE 0428-3-2006, D-Cinema Distribution Master – Audio Channel Mapping and Channel Labeling

Annex B: Synchronization (informative)

B.1 Synchronization issues

The following synchronization issues have been identified:

1) Rate matching between the transmitter and receiver

2) Adjusting the presentation time at a receiver

3) Adjusting the location at a transmitter

The rate matching between the transmitter and receiver can be done by one of two methods:

1) Clock-based rate control

2) Command-based rate control (see clause 10.4 Command based rate control mode (FDF = $00001xxx_2$)).

Clock-based rate control may use sampling clock delivery in an isochronous stream or another clock delivery system such as a dedicated clock.

The presentation time adjustment of the application sequence at a receiver can be done since the time stamp of a CIP is defined such that it reflects the time when the corresponding audio sample goes out of a buffer for depacketization. If an application requires precise adjustment of the presentation time, the application should take into account the extra delay caused by signal processing or A/D and D/A conversion.

B.2 Delivery of sampling clock of arbitrary frequency

This section focuses on rate matching in terms of sampling clock delivery, which is very familiar to audio engineers. It only applies for a real time transfer, which occurs when the sample transmission frequency is used to define the sampling frequency.

Since a CIP without a source packet header (SPH) has only one time stamp in the SYT field, the maximum synchronization clock frequency must be limited to the isochronous cycle of 8 kHz.

Assume that a transmitter carries an audio stream with sampling frequency *STF* and that *STF* > 8 kHz.

The transmitter derives a "synchronization clock" with frequency *Fsync* according to the equation B1:

Fsync = STF / SYT_INTERVAL < 8000 *(B1)*

where

Fsync is the synchronization clock frequency (in Hz);

STF is the sampling transmission frequency (in Hz);

SYT_INTERVAL denotes the number of events between two successive valid SYTs, which includes one of the events with a valid SYT;

The transmitter quantizes the timing of the "synchronization clock", for instance the rising edge of the clock, by referring to its own CYCLE_TIME. It transmits the sum of the timing and TRANSFER_DELAY by using the SYT field of the CIP. The resolution of the time stamp is 1/(24.576 MHz), or approximately 40ns, and CYCLE_TIME may have 40 ns of jitter due to this quantisation. If the timing information is not available for a CIP, the SYT must indicate the "No Information" code.

A receiver can reproduce the "synchronization clock" in terms of the pulse generated when the SYT equals its own CYCLE_TIME.

The sampling clock can be reproduced by multiplying the "synchronization clock" by the SYT_INTERVAL, which must be determined before receiving begins.

This sampling clock delivery does not require synchronization of the sampling clock and the isochronous cycle.

The reproduced synchronisation clock will have jitter. This jitter can degrade audio quality unless adequate jitter attenuation is used.

The local CYCLE_TIME registers at the transmitter and the receiver nodes will have jitter from various sources. This CYCLE_TIME register jitter has a minimum peak-peak amplitude equivalent to the approximately 40ns resolution of CYCLE_TIME. If one of the nodes is the cycle master this jitter only applies to CYCLE_TIME at the other node. If neither of the nodes is the cycle master then it will apply to CYCLE_TIME at both the transmit and received nodes. There is also a source of CYCLE_TIME jitter from the quantization of the correction for variable delay to the cycle start packets from the cycle master.

The jitter added to the synchronisation clock by delivery in this manner is the sum of the CYCLE_TIME jitter and the jitter due to the quantization of the time stamp.

Annex C: Catching up in Non-Blocking Transmission method (informative)

In clause 7.4.1 Non-Blocking transmission method, equation (3) provides that in normal operation, each transmitter shall construct a packet containing between 0 and SYT_INTERVAL events. Table 9-5 – Default SFC table specifies SYT_INTERVAL for each Sample transmission frequency such that:

Event_arrival_time[SYT_INTERVAL-1] – Event_arrival_time[0] > Min_period *(C2)*

And

125µs <= Min Period

where

Event_arrival_time[M] is the time (measured in µs) of the arrival at the transmitter of event at index M. The event with index = 0 is the event which has Presentation Time = SYT.

Min_Period is the time (measured in µs) of SYT_INTERVAL events.

The Min_Period ensures that at most only a single SYT will be required for each packet.

In the normal non-blocking transmission method, fewer than SYT_INTERVAL events will be transmitted in each packet. In the event of lost opportunities to transmit a packet (such as a cycle start packet drop after a Bus reset), a transmitter can catch up by transmitting up to SYT_INTERVAL events in one or more of the subsequent packets. Events, which are late according to equation (4) in clause 7.4.1 are not transmitted.

Equation (9) in clause 9.2 can be used to determine the required isochronous bandwidth, but in normal non-blocking operation, not all of this bandwidth is used. Extra bandwidth is available for catching up, however this extra bandwidth may not be sufficient to ensure that some events will not be late.

A method is provided below to allow a transmitter to add one extra event to each "catch-up packet" as long as the total number of events is not greater than SYT_INTERVAL.

In equation (9) in clause 9.2, the term (int(max(Fs)/8000) + 1) can be changed to (int(max(Fs)/8000) + 2). This increases the allocated bandwidth such that one additional event can be sent per packet. While this bandwidth will be unused during normal operation, it will provide the extra bandwidth needed to catch up without violating the allocated bandwidth.

It is important to consider that in the case of lost isochronous cycles, more than one transmitter may be trying to catch up at the same time. Sufficient bandwidth should be allocated to allow for catch-up.

Annex D: Transport characteristics (informative)

D.1 Sampling clock jitter characteristics

Sampling clock jitter can degrade the accuracy of conversion processes in sampling devices. This part of the annex describes the jitter mechanisms in the exchange of sample timing information and derives worst-case jitter levels to be used for stressing sampling devices when making performance measurements.

This issue applies to systems that require a sample clock to be transferred across the Bus to a sampling device. For example, it does not apply for devices that use flow control with a single sampling device acting as destination and synchronization master, or where the destination device is a non-sampling device such as a recorder.

D.1.1 Definitions

D.1.1.1 Sample clock

The reference used at a sampling device to define the instant at which an audio data sample word is valid. For oversampled conversion systems, the sample clock is multiplied up to the oversampling rate. Inside an asynchronous sampling frequency converter (ASFC), one sample clock is represented numerically by the relationship it has to another sample clock.

D.1.1.2 Sampling frequency, Fs

This is the frequency of the sample clock.

D.1.1.3 Sample clock timing transfer

This is the mechanism by which the sample clock of one device can be derived from a clock on another device such as by using an embedded synchronization clock.

D.1.1.4 Embedded synchronization clock

A signal that carries information that is used by a sampling device to derive a sample clock. In the context of A/M protocol, this synchronization clock is embedded in the SYT field of the CIP and carries timing information that refers to local CYCLE_TIME register values.

D.1.1.5 Synchronization clock frequency, Fsync

The embedded synchronization clock frequency using the A/M protocol has to be less than the isochronous cycle rate of 8kHz. The rate is defined as the following:

Fsync = Fs / SYT_INTERVAL

The SYT_INTERVAL value is defined in the CIP header for each sampling frequency.

D.1.1.6 Sampling device

A device that depends on the timing of a sample clock to modify an audio signal in some way as it is being converted between the analog and digital domains, or between two independent sampling frequencies. Examples of a sampling device are an analog to digital converter (ADC), a digital to analog converter (DAC) and an ASFC.

D.1.1.7 Non-sampling device

Devices that do not use clock timing in a way that may modify the analog or digital audio signal. Any clocks that they use do not affect the accuracy of that data in normal operation. (Compare with sampling device).

D.1.1.8 Synchronization clock source

A device that supplies an embedded synchronization clock that another device uses to derive a sample clock. This does not need to be a source device for audio data.

D.1.1.9 Synchronization clock destination

D.1.1.10 Clock jitter

This is the deviation in the timing of clock transitions when compared with an ideal clock. The ideal clock can be considered to have a frequency of exactly the same long-term average frequency and aligned for zero mean phase offset from the real clock. For a sample clock, the jitter amplitude defined in this way is directly related to the amplitude of the jitter modulation products produced in a sampling device.

D.1.1.11 Embedded synchronization clock jitter

Jitter in the embedded synchronization clock includes the effect of errors (including limited precision) in the embedded SYT data and jitter in the CYCLE_TIME register used to decode the SYT.

D.1.2 Sample clock transfer jitter mechanisms using A/M protocol

The A/M protocol and the Serial Bus use asynchronous clocks to define and exchange timing and synchronization information. The changing phase relationships and limited timing resolution of these clocks, and in some circumstances, the changing phase relationship to an external sample clock, produce a variable error which introduces jitter into an embedded synchronization clock.

There are other sources of jitter including oscillator phase noise, variable gate delays and cable inter-symbol interference. These are normally small in comparison with the mechanisms considered here.

D.1.2.1 CYCLE_TIME register jitter

Embedded synchronization clock information is referenced to the CYCLE_TIME register value at the synchronization clock source. Jitter on this register value at the synchronization clock source and synchronization clock destination nodes contributes to embedded synchronization clock jitter.

D.1.2.1.1 Cycle start packet CYCLE_TIME resolution

The cycle start packet issued from the cycle master is used to align the CYCLE_TIME registers of any isochronous-capable nodes on a Serial Bus. It is transmitted at or after cycle counter on the cycle master node is incremented. It carries the value of the cycle master node CYCLE_TIME register at the time the cycle start is initiated.

Asynchronous activity on the Bus at the time of the cycle starts event causes a delay in transmitting the cycle start packet. At the other isochronous nodes, the CYCLE_TIME register is loaded with the value carried on the cycle start packet. That compensates for the cycle start delay but only up to the resolution of that register. This resolution is 1/24.576MHz (which is approximated in this annex as 41ns).

The cycle start packet carries a value from the CYCLE_TIME register. If the transmission of the packet is timed so that it always occurs at a fixed time after the moment that the CYCLE_TIME register is updated to that value, then cycle start delays will be corrected without significant error. This means that asynchronous activity at the time of the cycle start event will not be a source of jitter.

However, some IEEE1394 compliant implementations might introduce a variable delay between the time the CYCLE_TIME register is updated and cycle start packet transmission of that value. This will depend on the implementation but this delay may be limited to less than the 41ns CYCLE_TIME resolution or it could possibly be even greater than this.

D.1.2.1.2 Variable transport delay to cycle start packets

As a cycle start packet is passed through intermediate nodes on the Bus it is delayed by a variable amount of repeater data delay.

The normal mechanism for the variation in this delay is the re-timing of the packet by the local clock at each node. The repeater data delay varies as the relative timing of the incoming transitions and the local clock changes. This change is a result of the frequency difference between the local clock and the clock on the previous node the packet has passed through. Jitter produced in this way is in the form of a ramping variation with a step correction in the opposite direction. The frequency of this 'sawtooth' is related to the frequency difference between the two node clocks.

IEEE1394 does not define explicit limits for repeater delay jitter. The draft supplement, P1394a, specifies a PHY register field 'Jitter' that can indicate values from 1/49.152MHz (which is approximated in this annex as 20ns) to 7/49.152MHz (approximately 163ns).

IEEE1394 PHY devices that resynchronize received data with a 49.152MHz clock will have repeater data delay jitter approximately 20ns peak-peak or 6ns RMS.

The jitter due to variable repeater delay jitter is cumulative. The total variable transport delay is the sum of the delay at each node. The total RMS jitter to the cycle start packet transport delay is the root sum of squares (RSS) of the RMS jitter at each intermediate repeater node.

D.1.2.1.3 Quantization of CYCLE_TIME register correction

The CYCLE_TIME registers at each isochronous node increment at a rate defined by the exact rate of the 24.576MHz clock in the local node. These registers are time aligned with similar registers in other nodes by being loaded with the value carried in the cycle start packet transmitted by the cycle master. As the CYCLE_TIME register incrementing clock has a slightly different frequency at each node, there will be a gradually changing error between the updating of that register at the cycle master and the other nodes.

When there is a difference between the value on an incoming cycle start packet and the value in the local CYCLE_TIME register then a correction is made.

This correction is quantized to the CYCLE_TIME register resolution of 1/24.576MHz. The contribution of this mechanism to the CYCLE_TIME register jitter is normally a gradually increasing delay or advance with corrective step in the opposite direction. This jitter has an amplitude equivalent to the CYCLE_TIME resolution of 41ns peak to peak and 12ns RMS.

D.1.2.2 Time-stamp quantization jitter

The time stamp (SYT) carrying the sampling timing information has a resolution of 1/24.576MHz. The effect of quantization to this resolution is to add jitter to the embedded sample clock. This jitter has an amplitude equivalent to the SYT resolution of 41ns peak to peak and 12ns RMS. It will have frequency components related to the beat frequency between the time stamp rate (Fs/SYT_INTERVAL) and the 24.576MHz clock incrementing the CYCLE_TIME register.

D.1.3 Embedded sample clock jitter

D.1.3.1 Embedded sample clock jitter spectrum

The error in the values and timing of the embedded synchronization clock can be considered as a time-varying signal. This can be examined in the frequency domain through spectrum analysis. This jitter spectrum will relate to the jitter spectrum in the sample clock transfer mechanism and the jitter transfer function.

There are discrete frequency components corresponding to the fundamental and harmonic frequencies associated with each of the applicable jitter sources described in the previous clause. These frequencies depend on the frequency differences between the local PHY clocks on the nodes.

Any jitter source that produces a jitter signal similar to a sawtooth will have discrete jitter frequency components at the sawtooth frequency and multiples of that rate. Where the multiple is at a frequency above half the frequency that the timing information is updated, then that component will be aliased to below that rate and the signal will no longer appear as a sawtooth.

D.1.3.2 Embedded sample clock jitter amplitude

The total amount of embedded sample clock jitter is dependent on the following:

- The number of nodes between the cycle master and sample clock source.
- The number of nodes between the cycle master and sample clock destination.
- The implementation of each node.
- Whether or not the sample clock source is synchronized to the Bus.

D.1.3.2.1 Example One: Simple two-node Bus

As an example, examine the simplest two-node system. This has the cycle master as the sample clock source node (node 0), and the sample clock is locked to the sample clock source node PHY clock at a multiple of the cycle time rate. Asynchronous activity is low enough to ensure that the cycle start packet is never delayed.

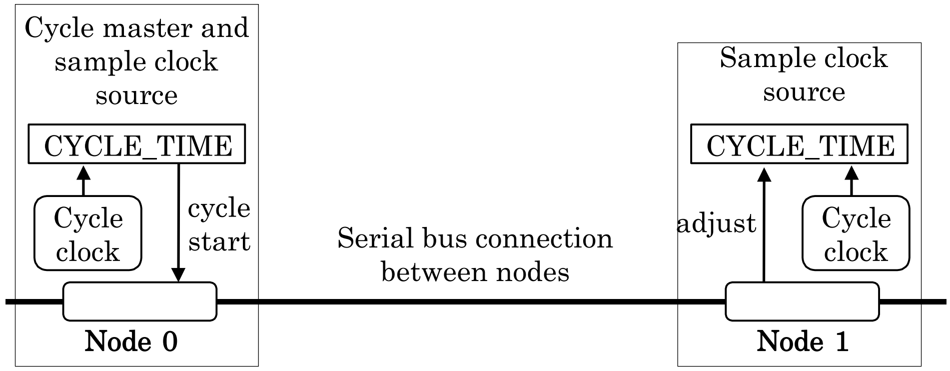

Figure D. 1 – Two-node Bus

- There will not be any jitter due to cycle start packet CYCLE_TIME resolution as the cycle start packet is not being delayed due to asynchronous activity.
- There is no variable transport delay to cycle start packets as there are no intermediate nodes on the Bus.
- Quantization of CYCLE_TIME register correction in the sample clock destination node will be a source of jitter in this example. This will be in the form of one sawtooth at a frequency determined by the offset between the cycle start rate and the sample clock destination PHY clock. This will have an amplitude of approximately 12ns RMS (41ns peak to peak).
- As the sample clock is frequency-locked to the cycle master PHY clock, there is no time-stamp quantization jitter.

Therefore, for the simple two-node system in this example, the recovered embedded sample clock will have just one systematic jitter source. This will have a jitter amplitude of approximately 12ns RMS (41ns peak to peak) in the form of a sawtooth at a rate determined by the frequency offset between the two PHY node clocks.

D.1.3.2.2 Example two: three-node Bus

For this example, there are three nodes which are separately the cycle master node, sample clock source node and sample clock destination node.

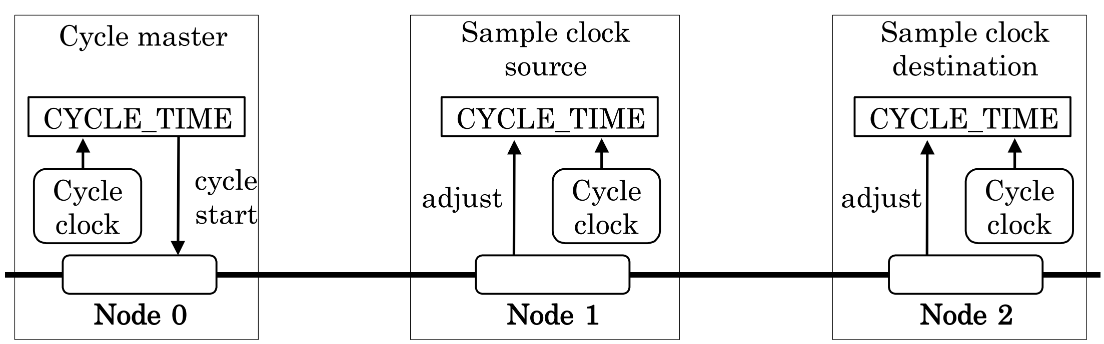

Figure D. 2 – Three-node Bus

The following analysis also assumes that the sample clock is not synchronous to any of the Bus clocks

— If the cycle start packet is sometimes delayed, there may be some jitter caused when the cycle start packet CYCLE_TIME value does not exactly correspond with the delay to the transmission of the packet. This will have a peak amplitude that is dependent on the implementation of the cycle master cycle start transmission mechanism. (The amplitude of this mechanism is not included in the analysis.)

— In the path from cycle master (node 0) to node 1 there are no intermediate nodes. In the path from cycle master (node 0) to node 2, there is one intermediate node that will have a variable transport delay to cycle start packets. This will contribute to the jitter in the CYCLE_TIME value at that node. This jitter will be in the form of a sawtooth related to the beating of the node 0 and node 1 cycle clocks. The amplitude of this jitter mechanism depends on the implementation of the repeater function in this node. This analysis assumes that this repeater includes resynchronization with a 49.152MHz clock. This will contribute jitter of approximately 6ns RMS (20ns peak to peak).

— Quantization of CYCLE_TIME register correction in nodes 1 and 2 will be a source of jitter. In each of these nodes, this will be in the form of a sawtooth at a frequency determined by the offset between the cycle start rate and the node PHY clock. These two sources of jitter will each have an amplitude of approximately 12ns RMS (41ns peak to peak).

— At node 1, the sample clock timing is encoded into the SYT with the resolution of the CYCLE_TIME register. The sample clock is asynchronous to the update of the CYCLE_TIME register. The error due to the variation in relative phase of the clocks is a sawtooth with a frequency determined by the difference between the node 1 cycle clock frequency and the time stamp rate. This source of jitter will have an amplitude of approximately 12ns RMS (41ns peak to peak).

This illustrates how this system has four sources of periodic jitter (excluding the source of jitter related to asynchronous activity): Three of 12ns RMS and one of 6ns RMS. The sum total of the periodic jitter (excluding the component due to asynchronous activity) will be 21ns RMS. (This would also have a peak to peak value of 132ns. This value represents the infrequent coincidence of the peaks of all the contributing jitter components and would be an infrequent occurrence.)

D.1.3.2.3 Example: thirty-five-node system

This example illustrates a large Bus configuration with 23 hops between the cycle master (node 0) and each sample clock source (node 23) and sample clock destination (node 34). (According to IEEE Std 1394a-2000 [R2], this configuration represents a maximum within the constraints of a maximum PHY delay of 144ns and maximum cable length of 4.5m.)

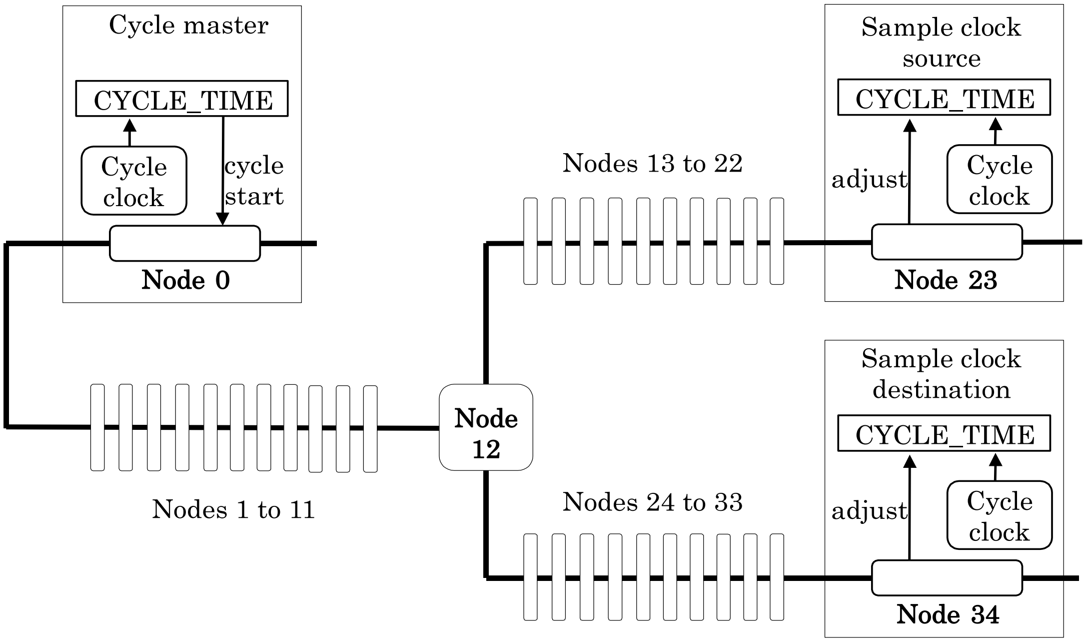

Figure D. 3 – Thirty-five-node Bus

The following analysis also makes similar assumptions for the 3-node example with respect to the sample clock.

- If there is asynchronous activity on the Bus, then the jitter mechanism due to cycle start packet delay is the same as for the three-node example. This is not included in the analysis.
- In the paths from the cycle master (node 0) to both the sample clock source (node 23) and sample clock destination (node 34) there are 22 intermediate nodes. Each of these will impose a variable transport delay on to cycle start packets in the same manner as the 3-node example. The peak jitter will scale in proportion to the number of hops (22) and the RMS jitter will scale with the square root of that number, 4.7. If each repeater applies re-synchronization with a local 49.152MHz clock, then they will add a total of 28ns RMS of jitter to the arrival time of the cycle start packet at the sample clock source (node 23) and at the sample clock destination (node 34).
- As with the 3-node example, quantization of CYCLE_TIME register correction at the sample clock source and destination will be a source of jitter of amplitude 12ns RMS each.
- As with the 3-node example, the time-stamp quantization jitter will add 12ns RMS.

This illustrates how this system has three sources of periodic sawtooth jitter at 12ns RMS and two summed periodic components at 28ns RMS each. The sum total of the periodic jitter is 44 ns RMS.

This result does not represent a 'worst case'. The variable transport delay jitter at each intermediate node could be significantly greater than 20ns while remaining compliant with IEEE1394. The potential variable error in the CYCLE_TIME value in the cycle start packet (when the cycle start has been delayed by asynchronous activity) has also not been included.

D.1.4 Jitter attenuation

This occurs with the filtering function of the sample clock recovery device. This will have a low pass jitter attenuation characteristic. Sample clock jitter causes modulation of the sampled signal. These modulation products may become audible. For high quality applications, it is recommended that the jitter attenuation characteristic of the sample clock recovery system satisfies the template shown in Figure D.4.

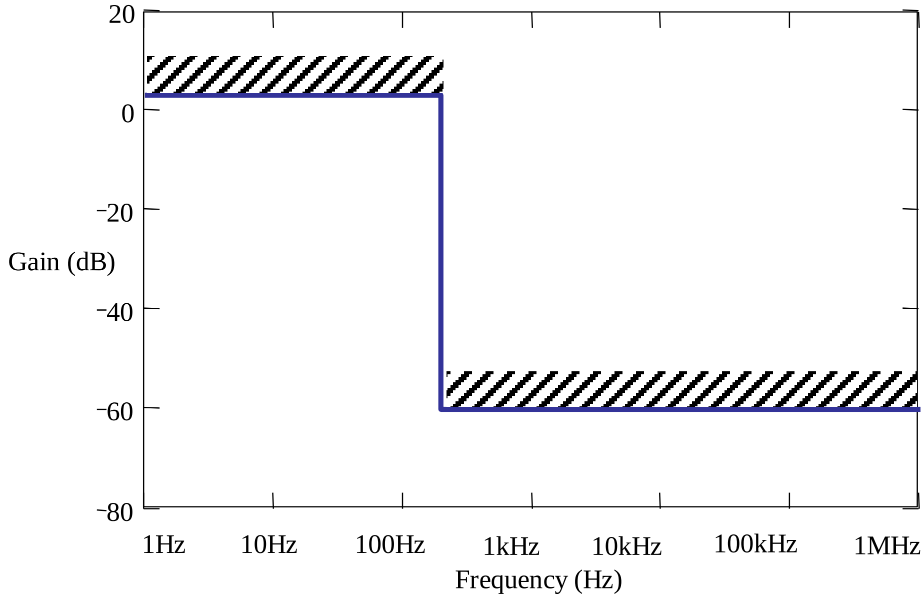

Figure D. 4 – Sample clock recovery jitter attenuation template

To satisfy this template, the jitter attenuation plotted against jitter frequency shall fall below the shaded clauses of the graph. The attenuation shall exceed 60dB at jitter frequencies above 200Hz and up to half the recovered sample clock frequency. Below 200Hz the gain shall not exceed 3dB.

The jitter attenuation for received jitter at frequencies, f_r above half the SYT_MATCH clock rate, f_s is determined by the response to the images of the received jitter that may be present in the sampling clock. These will be present at image frequencies of:

$$f_i = N \cdot f_s \pm f_r$$

Where N is an integer.

D.1.5 Jitter measurement

Jitter meters approximate the long-term average frequency and phase of a signal that they are measuring. This will result in a highs characteristic. As the sample clocks derived using the A/M protocol have a strong low frequency jitter component, the low frequency corner frequency of the jitter meter is important.

It is recommended that jitter measurements use the characteristics defined by the jitter measurement filter characteristic of Figure D.5.

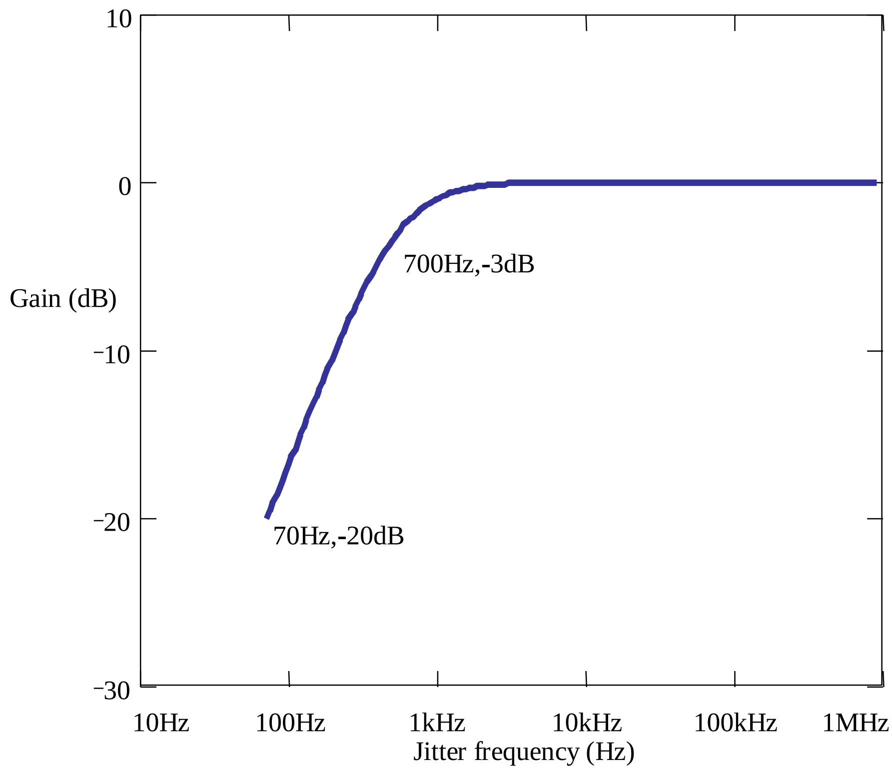

Figure D. 5 – Sample clock jitter measurement filter characteristic

This is a minimum- phase high pass filter with a -3 dB frequency of 700 Hz, a first order roll-off to 70 Hz and with a pass-band gain of unity. Note: This is compatible with the intrinsic jitter measurement filter characteristic used in IEC 60958-3 and -4.