DIN EN IEC 62680-1-5

DIN

ICS 33.120.01; 35.200 Einsprüche bis 2020-02-06

Entwurf

Schnittstellen des Universellen Seriellen Busses für Daten und Energie – Teil 1-5: Gemeinsame Komponenten – USB Audio 3.0 Geräteklassendefinition (IEC 100/3157/CDV:2018); Englische Fassung prEN IEC 62680-1-5:2018

Universal serial bus interfaces for data and power –
Part 1-5: Common components –
USB Audio 3.0 Device Class Definition
(IEC 100/3157/CDV:2018);
English version prEN IEC 62680-1-5:2018

Anwendungswarnvermerk

Dieser Norm-Entwurf mit Erscheinungsdatum 2019-12-06 wird der Öffentlichkeit zur Prüfung und Stellungnahme vorgelegt.

Weil die beabsichtigte Norm von der vorliegenden Fassung abweichen kann, ist die Anwendung dieses Entwurfs besonders zu vereinbaren.

Stellungnahmen werden erbeten

- vorzugsweise online im Norm-Entwurfs-Portal von DIN unter www.din.de/go/entwuerfe bzw. für Norm-Entwürfe der DKE auch im Norm-Entwurfs-Portal der DKE unter www.entwuerfe.normenbibliothek.de, sofern dort wiedergegeben;
- oder als Datei per E-Mail an dke@vde.com möglichst in Form einer Tabelle. Die Vorlage dieser Tabelle kann im Internet unter www.din.de/go/stellungnahmen-norm-entwuerfe oder für Stellungnahmen zu Norm-Entwürfen der DKE unter www.dke.de/stellungnahme abgerufen werden;
- oder in Papierform an die DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE, Stresemannallee 15, 60596 Frankfurt am Main.

Die Empfänger dieses Norm-Entwurfs werden gebeten, mit ihren Kommentaren jegliche relevanten Patentrechte, die sie kennen, mitzuteilen und unterstützende Dokumentationen zur Verfügung zu stellen.

Gesamtumfang 156 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

AVA Documents Document Sharing Hub

Full document details: E DIN EN IEC 62680-1-5:2020-01

Nationales Vorwort

Die Englische Fassung des europäischen Dokuments prEN IEC 62680-1-5:2018 „Schnittstellen des Universellen Seriellen Busses für Daten und Energie – Teil 1-5: Gemeinsame Komponenten – USB Audio 3.0 Geräteklassendefinition“ (Entwurf in der Umfrage) ist unverändert in diesen Norm-Entwurf übernommen worden.

Die Internationale Elektrotechnische Kommission (IEC) und das Europäische Komitee für Elektrotechnische Normung (CENELEC) haben vereinbart, dass ein auf IEC-Ebene erarbeiteter Entwurf für eine Internationale Norm zeitgleich (parallel) bei IEC und CENELEC zur Umfrage (CDV-Stadium) und Abstimmung als FDIS (en: Final Draft International Standard) bzw. Schluss-Entwurf für eine Europäische Norm gestellt wird, um eine Beschleunigung und Straffung der Normungsarbeit zu erreichen. Dem entsprechend ist das internationale Dokument IEC 100/3157/CDV:2018 „Universal serial bus interfaces for data and power – Part 1-5: Common components – USB Audio 3.0 Device Class Definition“ unverändert in den Entwurf prEN IEC 62680-1-5:2018 übernommen worden.

Das internationale Dokument wurde vom TC 100 „Audio, video and multimedia systems and equipment“ der Internationalen Elektrotechnischen Kommission (IEC) erarbeitet.

Bei der Abstimmung zu dem Europäischen Schluss-Entwurf bei CENELEC und dem Internationalen Schluss-Entwurf bei IEC [Final Draft International Standard (FDIS)] sind jeweils nur „JA/NEIN“-Entscheidungen möglich, wobei „NEIN“-Entscheidungen fundiert begründet werden müssen. Dokumente, die bei CENELEC als Europäische Norm angenommen und ratifiziert werden, sind unverändert als Deutsche Normen zu übernehmen.

Für dieses Dokument ist das nationale Arbeitsgremium K 742 „Audio-, Video- und Multimediasysteme, -geräte und -komponenten“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf ein Dokument ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils aktuellste Ausgabe des in Bezug genommenen Dokuments.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe des Dokuments.

Der Zusammenhang der zitierten Dokumente mit den entsprechenden deutschen Dokumenten ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für den vorliegenden Normentwurf erfüllt.

Da sich die Benutzer des vorliegenden Norm-Entwurfs der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der IEC 62680-1-5 CDV veröffentlicht. Zu deren die Einleitung, Abschnitt 1.1, der den Anwendungsbereich festlegt, und Abschnitt 1.4, der die Begriffe und Abkürzungen festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA des vorliegenden Norm-Entwurfs hinzugefügt. Für einige der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Dieser Norm-Entwurf steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Es ist vorgesehen, auch bei der entsprechenden zukünftigen Deutschen Norm auf die deutsche Sprachfassung zu verzichten und diese in der englischsprachigen Fassung zu veröffentlichen.

Es wird auf die Möglichkeit hingewiesen, dass einige Elemente dieses Dokuments Patentrechte berühren können. IEC ist nicht dafür verantwortlich, einige oder alle diesbezüglichen Patentrechte zu identifizieren.

Documents
Document Sharing Hub

Full document details: E DIN EN IEC 62680-1-5:2020-01

Das Original-Dokument enthält Bilder in Farbe, die in der Papierversion in einer Graustufen-Darstellung wiedergegeben werden. Elektronische Versionen dieses Dokuments enthalten die Bilder in der originalen Farbdarstellung.

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

– Leerseite –

Documents
Document Sharing Hub

Full document details: E DIN EN IEC 62680-1-5:2020-01

Schnittstellen des Universellen Seriellen Busses für Daten und Energie – Teil 1-5: Gemeinsame Komponenten – USB Audio 3.0 Geräteklassendefinition

Inhalt

Seite

Nationaler Anhang NA (informativ) Auszugsweise Übersetzung 2

Einleitung 2

1.1 Anwendungsbereich 2

1.4 Begriffe und Abkürzungen 3

Documents
Document Sharing Hub

Full document details: E DIN EN IEC 62680-1-5:2020-01

— Entwurf —

Nationaler Anhang NA
(informativ)

Auszugsweise Übersetzung

Reihenfolge und Inhalt der folgenden Abschnitte sind identisch mit denen in den jeweiligen Abschnitten der Englischen Fassung.

Einleitung

Die Reihe IEC 62680 beruht auf einer Reihe von Spezifikationen, die ursprünglich vom USB Implementers Forum (USB-IF) entwickelt wurden. Diese Spezifikationen wurden der IEC im Rahmen einer Sondervereinbarung zwischen der IEC und dem USB-IF überlassen.

Diese Norm ist die Veröffentlichung USB Device Class Definition for Audio Devices Release 3.0 des USB-IF.

Das USB Implementers Forum, Inc. (USB-IF) ist ein Non-Profit-Unternehmen, das von den Unternehmen gegründet wurde, die die Universal-Serial-Bus-Spezifikation entwickelt haben. Ziel der Gründung des USB-IF war es, ein/e Support-Organisation/-Forum für die Weiterentwicklung und Anwendung der Universal-Serial-Bus-Technologie zu schaffen. Das Forum fördert die Entwicklung von hochwertigen kompatiblen USB-Peripheriegeräten und zeigt die Vorteile von USB und der Qualität von Produkten auf, die die Konformitätsprüfungen bestanden haben.

JEDWEDE USB-SPEZIFIKATIONEN WERDEN OHNE ANPASSUNG WIEDERGEGEBEN, OHNE GARANTIEN GLEICH WELCHER ART, INSBESONDERE OHNE GARANTIEN DER MARKTGÄNGIGKEIT, DER NICHTVERLETZUNG UND DER EIGNUNG FÜR EINEN BESTIMMTEN ZWECK. DAS USB IMPLEMENTERS FORUM UND DIE VERFASSER DER USB-SPEZIFIKATIONEN ÜBERNEHMEN KEINERLEI HAFTUNG, INSBESONDERE KEINE HAFTUNG FÜR VERLETZUNGEN VON EIGENTUMSRECHTEN IM ZUSAMMENHANG MIT DER VERWENDUNG, DER IMPLEMENTIERUNG ODER DEN ANGABEN IN DIESER SPEZIFIKATION.

DIE BEREITSTELLUNG VON USB-SPEZIFIKATIONEN VERLEIHT IHNEN KEINERLEI LIZENZ, WEDER AUSDRÜCKLICH NOCH STILLSCHWEIGEND, DURCH DULDUNG ODER ANDERWEITIG, FÜR GEISTIGE EIGENTUMSRECHTE.

Durch das Abschließen eines sog. USB Adopters Agreement (USB-Anwendervereinbarung) kann das unterzeichnende Unternehmen jedoch gegebenenfalls eine gegenseitige RAND-Z-Lizenzvereinbarung für konforme Produkte eingehen. Für weitere Informationen siehe:

https://www.usb.org/documents

IEC BEZIEHT IN KEINER WEISE STELLUNG ZU DER FRAGE, OB ES RATSAM FÜR SIE IST, EIN USB ADOPTERS AGREEMENT ABZUSCHLIESSEN ODER SICH AM USB IMPLEMENTERS FORUM ZU BETEILIGEN.

1.1 Anwendungsbereich

Die Festlegung der Audiogeräteklasse gilt für alle in zusammengesetzte Geräte eingebetteten Geräte und Funktionen, die zur Bearbeitung von audio-, video- oder klangbezogenen Funktionalitäten dienen. Dies umfasst sowohl (analoge und digitale) Audiodaten als auch die Funktionalität, mit der die Audioumgebung direkt gesteuert wird, d. h. beispielsweise die Lautstärke- oder Klangsteller. Nicht zur Audiogeräteklasse zählen Funktionalitäten zum Betrieb von Transportmechanismen im Zusammenhang mit der Wiedergabe von Audiodaten, wie z. B. Band-Transportmechanismen oder die Steuerung eines CD-ROM-Laufwerks.

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

1.4 Begriffe und Abkürzungen

Für die Anwendung dieses Dokuments gelten die folgenden Begriffe. Für weitere Begriffe zum Universal Serial Bus siehe auch Kapitel 2 „Terms and Abbreviations“ (Begriffe und Abkürzungen) in der *USB Specification.*

ASRC	Abkürzung für Asynchronous Sampling Rate Conversion (asynchrone Abtastratenkonvertierung)
(Audiokanal-)Gruppe	Gruppe aus logischen Audiokanälen, die eng verwandte synchrone Audioinformationen übertragen. Ein Stereo-Audiodatenstrom ist ein typisches Beispiel für eine Gruppe aus zwei Kanälen.
Audiostellerattribut	Parameter eines Audiostellers. Beispiele sind die Attribute Aktuell, Minimum, Maximum und Auflösung eines Lautstärkestellers.
Audiosteller	Logisches Objekt, mit dem eine bestimmte Audioeigenschaft verändert wird. Beispiele sind Lautstärkesteller, Stummsteller usw.
Audiodatenstrom	Transportmedium, das Audioinformationen übertragen kann
AEC	Abkürzung für Audio Echo Cancellation (Audioechokompensation)
Audiofunktion	Unabhängiger Teil eines USB-Geräts, der für audiobezogene Funktionalitäten zuständig ist
Audioschnittstellen-Kombination (AIA, en: audio interface association)	Gruppe aus einer Audio-Control-Schnittstelle, keiner oder mehr Audio-Streaming-Schnittstellen und keiner oder mehr MIDI-Streaming-Schnittstellen, die zusammen eine gemeinsame Schnittstelle zu einer bestimmten Version (Konformitätsstufe) der Audiofunktion bilden
Audio-Control-Schnittstelle (ACI, en: AudioControl interface)	USB-Schnittstelle für den Zugriff auf die Audiosteller in einer Audiofunktion
Audio-Streaming-Schnittstelle (ASI, en: AudioStreaming interface)	USB-Schnittstelle für die Übertragung von Audiodatenströmen in eine Audiofunktion hinein oder aus einer Audiofunktion heraus
Taktmultiplikator (CM, en: clock multiplier)	Multipliziert ein Taktsignal mit einem Teilwertmultiplikator
Taktwähler (CX, en: clock selector)	Trifft eine Auswahl aus einer Reihe von Eingangstaktsignalen
Taktquelle (CS, en: clock source)	Erzeugt ein audiobezogenes Taktsignal (Zentraltakt oder Abtasttakt)
CMD	Abkürzung für Clock Multiplier Descriptor (Taktmultiplikator-Deskriptor)
CSD	Abkürzung für Clock Source Descriptor (Taktquellen-Deskriptor)
CXD	Abkürzung für Clock Selector Descriptor (Taktwähler-Deskriptor)
Effektgerät (EU, en: effect unit)	Ermöglicht eine umfangreiche Veränderung der Audiosignale auf den logischen Eingangsaudiokanälen
Entität	Adressierbares logisches Objekt innerhalb einer Audiofunktion
Erweiterungseinheit (XU, en: extension unit)	Wendet einen undefinierten Prozess auf eine Reihe logischer Eingangskanäle an

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

Feature-Gerät (FU, en: feature unit)	Ermöglicht eine grundlegende Veränderung der Audiosignale auf den logischen Eingangsaudiokanälen
FUD	Abkürzung für Feature Unit Descriptor (Feature-Gerät-Deskriptor)
Hochleistungsdeskriptor	Neue Repräsentation für klassenspezifische Deskriptoren, die potenziell große (>256 Byte) und dynamische Deskriptoren ermöglicht
Eingangspin	Logische Eingangsverbindung zu einer Entität. Überträgt eine einzelne Gruppe.
Eingangsanschluss (IT, en: input terminal)	Buchse für eingehende Audioinformationen für die Audiofunktion
ITD	Abkürzung für Input Terminal Descriptor (Eingangsanschluss-Deskriptor)
Logischer Audiokanal	Logisches Transportmedium für einen einzelnen Audiokanal. Abstrahiert von den physischen Eigenschaften und Formaten der Verbindung. Wird üblicherweise über die räumliche Anordnung identifiziert. Beispiele sind Kanal links, Surround-Kanal rechts usw.
MIDI-Streaming-Schnittstelle (MSI, en: MIDIStreaming interface)	USB-Schnittstelle, die für die Übertragung von MIDI-Datenströmen in die Audiofunktion hinein oder aus der Audiofunktion heraus verwendet werden kann
Mischer (MU, en: mixing unit)	Führt eine Reihe logischer Eingangskanäle auf einer Reihe logischer Ausgangskanäle zusammen
MUD	Abkürzung für Mixer Unit Descriptor (Mischer-Deskriptor)
OTD	Abkürzung für Output Terminal Descriptor (Ausgangsanschluss-Deskriptor)
Ausgangspin	Logische Ausgangsverbindung zu einer Entität. Überträgt eine einzelne Gruppe.
Ausgangsanschluss (OT, en: output terminal)	Ausgang für ausgehende Audioinformationen der Audiofunktion
Phasenregelschleife	Hardware- oder Software-Steuersystem, das ein Ausgangssignal erzeugt, dessen Phase an die Phase eines Eingangssignals gebunden ist. Kommt üblicherweise bei der Taktrückgewinnung zum Einsatz.
Verarbeitungseinheit (PU, en: processing unit)	Wendet einen vordefinierten Prozess auf eine Reihe logischer Eingangskanäle an
PUD	Abkürzung für Processing Unit Descriptor (Verarbeitungseinheit-Deskriptor)
Reserviert	Schlüsselwort, das reservierte Bits, Bytes, Wörter, Felder und Codewerte bezeichnet, die für eine zukünftige Normierung vorgemerkt werden. Ihre Verwendung und Interpretation darf in zukünftigen Erweiterungen dieser Norm festgelegt werden und, soweit nicht anders angegeben, dürfen sie nicht in der Anbieterimplementierung verwendet oder verändert werden. Ein reserviertes Bit, Byte, Wort oder Feld muss vom Absender auf null gesetzt und vom Empfänger ignoriert werden.
RUD	Abkürzung für Sampling Rate Converter Unit Descriptor (Abtastratenkonvertierer-Deskriptor)
Abtastratenkonvertierer (RU, en: sampling rate converter	Wandelt den eingehenden Audiodatenstrom, der mit einer Abtastrate läuft, die sich synchron zu einem ersten Zentraltakt

Documents
Document Sharing Hub

Full document details: E DIN EN IEC 62680-1-5:2020-01

unit)	verhält, in einen ausgehenden Audiodatenstrom um, der mit einer Abtastrate läuft, die sich synchron zu einem zweiten Abtasttakt verhält, welcher sich wiederum unabhängig von dem ersten Zentraltakt verhält.
Wähler (SU, en: selector unit)	Trifft eine Auswahl aus einer Reihe von Gruppen
SUD	Abkürzung für Selector Unit Descriptor (Wähler-Deskriptor)
Anschluss	Adressierbares logisches Objekt innerhalb einer Audiofunktion, das eine Verbindung zur „Außenwelt“ der Audiofunktion repräsentiert.
Einheit	Adressierbares logisches Objekt innerhalb einer Audiofunktion, das eine bestimmte Audio-Unterfunktionalität repräsentiert.
XUD	Abkürzung für Extension Unit Descriptor (Erweiterungseinheit-Deskriptor)

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

UNIVERSAL SERIAL BUS INTERFACES FOR DATA AND POWER

Part 1-5: Common components – USB Audio 3.0 Device Class Definition

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

This document was developed by USB-IF. IEC cannot make any changes based on the agreement between IEC and USB-IF.
Note to layout: please leave/place figure titles above the figures

International Standard IEC 62680-1-5 has been prepared by technical area 18: Multimedia home systems and applications for end-user networks, of IEC technical committee 100: Audio, video and multimedia systems and equipment.

The text of this standard was prepared by the USB Implementers Forum (USB-IF). The structure and editorial rules used in this publication reflect the practice of the organization which submitted it.

The text of this International Standard is based on the following documents:

FDIS	Report on voting
XX/XX/FDIS	XX/XX/RVD

Full information on the voting for the approval of this International Standard can be found in the report on voting indicated in the above table.

Documents
Document Sharing Hub

© USB-IF:1997-2016

This document has been drafted in accordance with the ISO/IEC Directives, Part 2.

The committee has decided that the contents of this document will remain unchanged until the stability date indicated on the IEC website under "http://webstore.iec.ch" in the data related to the specific document. At this date, the document will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

The National Committees are requested to note that for this document the stability date is ….

THIS TEXT IS INCLUDED FOR THE INFORMATION OF THE NATIONAL COMMITTEES AND WILL BE DELETED AT THE PUBLICATION STAGE.

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

INTRODUCTION

The IEC 62680 series is based on a series of specifications that were originally developed by the USB Implementers Forum (USB-IF). These specifications were submitted to the IEC under the auspices of a special agreement between the IEC and the USB-IF.

This standard is the USB-IF publication USB Device Class Definition for Audio Devices Release 3.0.

The USB Implementers Forum, Inc.(USB-IF) is a non-profit corporation founded by the group of companies that developed the Universal Serial Bus specification. The USB-IF was formed to provide a support organization and forum for the advancement and adoption of Universal Serial Bus technology. The Forum facilitates the development of high-quality compatible USB peripherals (devices), and promotes the benefits of USB and the quality of products that have passed compliance testing.

ANY USB SPECIFICATIONS ARE PROVIDED TO YOU "AS IS, "WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE. THE USB IMPLEMENTERS FORUM AND THE AUTHORS OF ANY USB SPECIFICATIONS DISCLAIM ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY PROPRIETARY RIGHTS, RELATING TO USE OR IMPLEMENTATION OR INFORMATION IN THIS SPECIFICAITON.

THE PROVISION OF ANY USB SPECIFICATIONS TO YOU DOES NOT PROVIDE YOU WITH ANY LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS.

Entering into USB Adopters Agreements may, however, allow a signing company to participate in a reciprocal, RAND-Z licensing arrangement for compliant products. For more information, please see:

https://www.usb.org/documents

IEC DOES NOT TAKE ANY POSITION AS TO WHETHER IT IS ADVISABLE FOR YOU TO ENTER INTO ANY USB ADOPTERS AGREEMENTS OR TO PARTICIPATE IN THE USB IMPLEMENTERS FORUM."

AVA Documents
Document Sharing Hub

Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

1

UNIVERSAL SERIAL BUS
DEVICE CLASS DEFINITION
FOR
AUDIO DEVICES

6 **Release 3.0**

7 **September 22, 2016**

8

Documents
Document Sharing Hub

Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

SCOPE OF THIS RELEASE

This document is the Release 3.0 of this Device Class Definition.

CONTRIBUTORS

Joe Scanlon	Advanced Micro Devices
Rhoads Hollowell	Apple Inc.
Girault Jones	Apple Inc.
Matthew X. Mora	Apple Inc.
Tzung-Dar Tsai	C-Media Electronics, Inc.
Brad Lambert	Cirrus Logic, Inc.
Dan Bogard	Conexant Systems, Inc.
Pete Burgers	DisplayLink (UK), Ltd.
David Roh	Dolby Laboratories, Inc.
Leng Ooi	Google, Inc.
Pierre-Louis Bossart	Intel Corporation
David Hines	Intel Corporation
Abdul Rahman Ismail (Co-Chair)	Intel Corporation
Devon Worrell	Intel Corporation
Chandrashekhar Rao	Logitech, Inc.
Terry Moore	MCCI Corporation
Alex Lin	MediaTek, Inc.
Bala Sivakumar	Microsoft Corporation
Geert Knapen (Co-Chair & Editor)	NXP Semiconductors **PL Mobile Audio** 411 E. Plumeria drive San Jose, CA 95134, USA E-mail: geert.knapen@nxp.com
James Goel	Qualcomm, Inc.
Andre Schevciw	Qualcomm, Inc.
Jin-Sheng Wang	Qualcomm, Inc.
Morten Christiansen	Synopsys

REVISION HISTORY

Revision	Date	Filename	Description
1.0	Mar. 18, 98	Audio10.pdf	Release 1.0
2.0	May. 31, 06	Audio20 final.pdf	Release 2.0
3.0	Sep. 22, 16	Audio30.pdf	Release 3.0

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

Copyright © 1997-2016 USB Implementers Forum, Inc.
All rights reserved.

INTELLECTUAL PROPERTY DISCLAIMER

A LICENSE IS HEREBY GRANTED TO REPRODUCE THIS SPECIFICATION FOR INTERNAL USE ONLY. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, IS GRANTED OR INTENDED HEREBY.

USB-IF AND THE AUTHORS OF THIS SPECIFICATION EXPRESSLY DISCLAIM ALL LIABILITY FOR INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS RELATING TO IMPLEMENTATION OF INFORMATION IN THIS SPECIFICATION. USB-IF AND THE AUTHORS OF THIS SPECIFICATION ALSO DO NOT WARRANT OR REPRESENT THAT SUCH IMPLEMENTATION(S) WILL NOT INFRINGE THE INTELLECTUAL PROPERTY RIGHTS OF OTHERS.

THIS SPECIFICATION IS PROVIDED “AS IS” AND WITH NO WARRANTIES, EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE. ALL WARRANTIES ARE EXPRESSLY DISCLAIMED. USB-IF, ITS MEMBERS AND THE AUTHORS OF THIS SPECIFICATION PROVIDE NO WARRANTY OF MERCHANTABILITY, NO WARRANTY OF NON-INFRINGEMENT, NO WARRANTY OF FITNESS FOR ANY PARTICULAR PURPOSE, AND NO WARRANTY ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE.

IN NO EVENT WILL USB-IF, MEMBERS OR THE AUTHORS BE LIABLE TO ANOTHER FOR THE COST OF PROCURING SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA OR ANY INCIDENTAL, CONSEQUENTIAL, INDIRECT, OR SPECIAL DAMAGES, WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, ARISING IN ANY WAY OUT OF THE USE OF THIS SPECIFICATION, WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

NOTE: VARIOUS USB-IF MEMBERS PARTICIPATED IN THE DRAFTING OF THIS SPECIFICATION. CERTAIN OF THESE MEMBERS MAY HAVE DECLINED TO ENTER INTO A SPECIFIC AGREEMENT LICENSING INTELLECTUAL PROPERTY RIGHTS THAT MAY BE INFRINGED IN THE IMPLEMENTATION OF THIS SPECIFICATION. PERSONS IMPLEMENT THIS SPECIFICATION AT THEIR OWN RISK.

Dolby™, AC-3™, Pro Logic™ and Dolby Surround™ are trademarks of Dolby Laboratories, Inc.
All other product names are trademarks, registered trademarks, or service marks of their respective owners.

Please send comments via electronic mail to audio-chair@usb.org

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

74 # TABLE OF CONTENTS

75 Scope of This Release 2
76 Contributors 2
77 Revision History 2
78 Table of Contents 4
79 List of Tables 8
80 List of Figures 10
81 1 Introduction 12
82 1.1 Scope 12
83 1.2 Purpose 12
84 1.3 Related Documents 12
85 1.4 Terms and Abbreviations 12
86 2 Management Overview 15
87 2.1 Overview of Key Differences between ADC v2.0 and v3.0 15
88 3 Functional Characteristics 17
89 3.1 Introduction 17
90 3.2 Basic Audio Device Definition 19
91 3.3 Backwards Compatibility 19
92 3.4 Audio Interface Association (AIA) and Interface Association Descriptor 19
93 3.4.1 Audio Function Class 20
94 3.4.2 Audio Function Subclass 20
95 3.4.3 Audio Function Protocol 20
96 3.5 Audio Interface Class 20
97 3.6 Audio Interface Subclass 20
98 3.7 Audio Interface Protocol 21
99 3.8 Audio Function Category 21
100 3.9 Clock Domains 22
101 3.10 Power Domains 22
102 3.11 Audio Synchronization Types 22
103 3.11.1 Asynchronous 22
104 3.11.2 Synchronous 22
105 3.11.3 Adaptive 22
106 3.11.4 Implications of the Different Synchronization Types 22
107 3.12 Inter Channel Synchronization 24
108 3.13 Audio Function Topology 24
109 3.13.1 Cluster 29
110 3.13.2 Input Terminal 29
111 3.13.3 Output Terminal 30
112 3.13.4 Mixer Unit 31
113 3.13.5 Selector Unit 31

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

114 3.13.6 Feature Unit 32
115 3.13.7 Sampling Rate Converter Unit 32
116 3.13.8 Effect Unit 33
117 3.13.9 Processing Unit 37
118 3.13.10 Extension Unit 39
119 3.13.11 Clock Entities 40
120 3.14 Operational Model 41
121 3.14.1 AudioControl Interface 42
122 3.14.2 AudioStreaming Interface 43
123 3.14.3 Clock Model 44
124 3.14.4 Power Domains Model 44
125 3.14.5 Additional Power Considerations and Requirements 46
126 3.14.6 Binding between Physical Buttons and Audio Controls 46
127 4 Descriptors 48
128 4.1 Standard Descriptors 48
129 4.2 Class-Specific Descriptors 48
130 4.2.1 Traditional Class-Specific Descriptors 49
131 4.2.2 High Capability Class-Specific Descriptors 49
132 4.3 Cluster Descriptor 50
133 4.3.1 Cluster Descriptor Header 51
134 4.3.2 Cluster Descriptor Block 51
135 4.3.3 Example Cluster descriptor 58
136 4.3.4 CEA-861.2 Channel Mapping 60
137 4.4 Physical versus Logical Cluster 60
138 4.4.1 Mapping between Physical and Logical Clusters 60
139 4.5 AudioControl Interface Descriptors 62
140 4.5.1 Standard AC Interface Descriptor 62
141 4.5.2 Class-Specific AC Interface Descriptor 63
142 4.6 AudioControl Endpoint Descriptors 91
143 4.6.1 AC Control Endpoint Descriptors 91
144 4.6.2 AC Interrupt Endpoint Descriptors 92
145 4.7 AudioStreaming Interface Descriptors 92
146 4.7.1 Standard AS Interface Descriptor 92
147 4.7.2 Class-Specific AS Interface Descriptor 93
148 4.7.3 Class-Specific AS Valid Frequency Range Descriptor 94
149 4.8 AudioStreaming Endpoint Descriptors 94
150 4.8.1 AS Isochronous Audio Data Endpoint Descriptors 94
151 4.8.2 AS Isochronous Feedback Endpoint Descriptor 96
152 4.9 Class-specific String descriptors 97
153 5 Requests 99
154 5.1 Standard Requests 99

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

5.2 Class-Specific Requests 99
5.2.1 AudioControl Requests 100
5.2.2 AudioStreaming Requests 123
5.2.3 Additional Requests 125
6 Interrupts 129
6.1 Interrupt Data Message 129
6.2 Interrupt Sources 130
Appendix A. Audio Device Class Codes 131
A.1 Audio Function Class Code 131
A.2 Audio Function Subclass Codes 131
A.3 Audio Function Protocol Codes 131
A.4 Audio Interface Class Code 131
A.5 Audio Interface Subclass Codes 131
A.6 Audio Interface Protocol Codes 132
A.7 Audio Function Category Codes 132
A.8 Audio Class-Specific Descriptor Types 132
A.9 Cluster Descriptor Subtypes 133
A.10 Cluster Descriptor Segment types 133
A.11 Channel Purpose Definitions 133
A.12 Channel Relationship Definitions 134
A.13 Ambisonic Component Ordering Convention Types 135
A.14 Ambisonic Normalization Types 136
A.15 Audio Class-Specific AC Interface Descriptor Subtypes 136
A.16 Audio Class-Specific AS Interface Descriptor Subtypes 137
A.17 Audio Class-Specific String descriptor Subtypes 137
A.18 Extended Terminal Segment Types 137
A.19 Effect Unit Effect Types 137
A.20 Processing Unit Process Types 138
A.21 Audio Class-Specific Endpoint Descriptor Subtypes 138
A.22 Audio Class-Specific Request Codes 138
A.23 Control Selector Codes 138
A.23.1 AudioControl Interface Control Selectors 138
A.23.2 Clock Source Control Selectors 138
A.23.3 Clock Selector Control Selectors 139
A.23.4 Clock Multiplier Control Selectors 139
A.23.5 Terminal Control Selectors 139
A.23.6 Mixer Control Selectors 139
A.23.7 Selector Control Selectors 139
A.23.8 Feature Unit Control Selectors 140
A.23.9 Effect Unit Control Selectors 140
A.23.10 Processing Unit Control Selectors 142

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

196 A.23.11 Extension Unit Control Selectors....142
197 A.23.12 AudioStreaming Interface Control Selectors....142
198 A.23.13 Endpoint Control Selectors....143
199 A.24 Connector Types....143
200
201

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

202 # LIST OF TABLES

203 Table 4-1: Traditional Class-Specific Descriptor Layout 49
204 Table 4-2: High Capability Class-Specific Descriptor Layout 50
205 Table 4-3: Cluster Descriptor Header 51
206 Table 4-4: Cluster Descriptor Segment 52
207 Table 4-5: End Segment 52
208 Table 4-6: Cluster Description Segment 53
209 Table 4-7: Vendor-defined Segment 53
210 Table 4-8: Channel Relationships 54
211 Table 4-9: Information Segment 57
212 Table 4-10: Ambisonic Segment 57
213 Table 4-11: Channel Description Segment 58
214 Table 4-12: Vendor-defined Segment 58
215 Table 4-13: Cluster Descriptor Example 58
216 Table 4-14: Standard AC Interface Descriptor 62
217 Table 4-15: Class-Specific AC Interface Header Descriptor 63
218 Table 4-16: Input Terminal Descriptor 65
219 Table 4-17: Output Terminal Descriptor 66
220 Table 4-18: Extended Terminal Descriptor Header 67
221 Table 4-19: Cluster Descriptor Segment 68
222 Table 4-20: End Segment 69
223 Table 4-21: Vendor-defined Segment 69
224 Table 4-22: Bandwidth Segment 70
225 Table 4-23: Magnitude Segment 70
226 Table 4-24: Magnitude/Phase Segment 70
227 Table 4-25: Position Segment 71
228 Table 4-26: Position Segment 71
229 Table 4-27: Vendor-defined Segment 72
230 Table 4-28: Connectors Descriptor 73
231 Table 4-29: Mixer Unit Descriptor 75
232 Table 4-30: Selector Unit Descriptor 76
233 Table 4-31: Feature Unit Descriptor 77
234 Table 4-32: Sampling Rate Converter Unit Descriptor 78
235 Table 4-33: Effect Unit Descriptor 79
236 Table 4-34: Parametric Equalizer Section Effect Unit Descriptor 80
237 Table 4-35: Reverberation Effect Unit Descriptor 81
238 Table 4-36: Modulation Delay Effect Unit Descriptor 81
239 Table 4-37: Dynamic Range Compressor Effect Unit Descriptor 82
240 Table 4-38: Common Part of the Processing Unit Descriptor 83
241 Table 4-39: Up/Down-mix Processing Unit Descriptor 84

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

242 Table 4-40: Stereo Extender Processing Unit Descriptor 85
243 Table 4-41: Multi-Function Processing Unit Descriptor 86
244 Table 4-42: Extension Unit Descriptor 87
245 Table 4-43: Clock Source Descriptor 88
246 Table 4-44: Clock Selector Descriptor 89
247 Table 4-45: Clock Multiplier Descriptor 90
248 Table 4-46: Power Domain Descriptor 91
249 Table 4-47: Standard AC Interrupt Endpoint Descriptor 92
250 Table 4-48: Standard AS Interface Descriptor 92
251 Table 4-49: Class-Specific AS Interface Descriptor 93
252 Table 4-50: Class-Specific AS Valid Frequency Range Descriptor 94
253 Table 4-51: Standard AS Isochronous Audio Data Endpoint Descriptor 95
254 Table 4-52: Class-Specific AS Isochronous Audio Data Endpoint Descriptor 96
255 Table 4-53: Standard AS Isochronous Feedback Endpoint Descriptor 96
256 Table 4-54: Class-specific String Descriptor 97
257 Table 5-1: Request Layout 101
258 Table 5-2: 1-byte Control CUR Parameter Block 102
259 Table 5-3: 1-byte Control RANGE Parameter Block 102
260 Table 5-4: 2-byte Control CUR Parameter Block 103
261 Table 5-5: 2-byte Control RANGE Parameter Block 103
262 Table 5-6: 4-byte Control CUR Parameter Block 104
263 Table 5-7: 4-byte Control RANGE Parameter Block 104
264 Table 5-8: INTEN Parameter Block 104
265 Table 5-9: Insertion Control CUR Parameter Block 107
266 Table 5-10: Band Numbers and Center Frequencies (ANSI S1.11-1986 Standard) 110
267 Table 5-11: Graphic Equalizer Control CUR Parameter Block 111
268 Table 5-12: Graphic Equalizer Control RANGE Parameter Block 111
269 Table 5-13: Valid Alternate Settings Control CUR Parameter Block 124
270 Table 5-14: Audio Data Format Control CUR Parameter Block 124
271 Table 5-15: Memory Request Values 125
272 Table 5-16: String Request 126
273 Table 5-17: High Capability Descriptor Request 127
274 Table 6-1: Interrupt Data Message Format 130
275 Table A-1: Audio Function Class Code 131
276 Table A-2: Audio Function Subclass Codes 131
277 Table A-3: Audio Function Protocol Codes 131
278 Table A-4: Audio Interface Class Code 131
279 Table A-5: Audio Interface Subclass Codes 131
280 Table A-6: Audio Interface Protocol Codes 132
281 Table A-7: Audio Function Category Codes 132
282 Table A-8: Audio Class-specific Descriptor Types 132

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

Table A-9: Audio Class-Specific Cluster Descriptor Subtypes 133
Table A-10: Cluster Descriptor Segment Types 133
Table A-11: Channel Purpose Definitions 133
Table A-12: Channel Relationship Definitions 134
Table A-13: Ambisonic Component Ordering Convention Types 135
Table A-14: Ambisonic Normalization Types 136
Table A-15: Audio Class-Specific AC Interface Descriptor Subtypes 136
Table A-16: Audio Class-Specific AS Interface Descriptor Subtypes 137
Table A-17: Audio Class-Specific String descriptor Subtypes 137
Table A-18: Extended Terminal Segment Types 137
Table A-19: Effect Unit Effect Types 137
Table A-20: Processing Unit Process Types 138
Table A-21: Audio Class-Specific Endpoint Descriptor Subtypes 138
Table A-22: Audio Class-Specific Request Codes 138
Table A-23: AudioControl Interface Control Selectors 138
Table A-24: Clock Source Control Selectors 138
Table A-25: Clock Selector Control Selectors 139
Table A-26: Clock Multiplier Control Selectors 139
Table A-27: Terminal Control Selectors 139
Table A-28: Mixer Control Selectors 139
Table A-29: Selector Control Selectors 139
Table A-30: Feature Unit Control Selectors 140
Table A-31: Reverberation Effect Unit Control Selectors 140
Table A-32: Reverberation Effect Unit Control Selectors 141
Table A-33: Modulation Delay Effect Unit Control Selectors 141
Table A-34: Dynamic Range Compressor Effect Unit Control Selectors 141
Table A-35: Up/Down-mix Processing Unit Control Selectors 142
Table A-36: Stereo Extender Processing Unit Control Selectors 142
Table A-37: Extension Unit Control Selectors 142
Table A-38: AudioStreaming Interface Control Selectors 142
Table A-39: Endpoint Control Selectors 143
Table A-40: Connector Types 143

LIST OF FIGURES

Figure 3-1: Multiple Configurations and their Audio Interface Associations and Interfaces 17
Figure 3-2: Audio Function Global View 18
Figure 3-3: Inside the Audio Function 28
Figure 3-4: Input Terminal Icon 30
Figure 3-5: Output Terminal Icon 31
Figure 3-6: Mixer Unit Icon 31

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

323 Figure 3-7: Selector Unit Icon 32
324 Figure 3-8: Feature Unit Icon 32
325 Figure 3-9: Sampling Rate Converter Unit Icon 33
326 Figure 3-10: PEQS Effect Unit Icon 34
327 Figure 3-11: Reverberation Effect Unit Icon 35
328 Figure 3-12: Modulation Delay Effect Unit Icon 35
329 Figure 3-13: Dynamic Range Compressor Transfer Characteristic 36
330 Figure 3-14: Dynamic Range Compressor Effect Unit Icon 37
331 Figure 3-15: Up/Down-mix Processing Unit Icon 38
332 Figure 3-16: Stereo Extender Processing Unit Icon 38
333 Figure 3-17 Multi-Function Processing Unit Icon 39
334 Figure 3-18: Extension Unit Icon 40
335 Figure 3-19: Clock Source Icon 41
336 Figure 3-20: Clock Selector Icon 41
337 Figure 3-21: Clock Multiplier Icon 41
338 Figure 4-1: Cluster Descriptor 51
339 Figure 4-2: Cluster Descriptor Block 52
340 Figure 4-3: 3D Representation of the Channel Relationships 56
341 Figure 4-4: Physical to Logical Cluster Mapping 61
342 Figure 4-5: Logical to Physical Cluster Mapping 62
343 Figure 4-6: Extended Terminal Descriptor 67
344 Figure 4-7: Extended Terminal Channel Block 68
345 Figure 4-8: Mixer internals 75
346

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

1 INTRODUCTION

1.1 SCOPE

The Audio Device Class Definition applies to all devices or functions embedded in composite devices that are used to manipulate audio, voice, and sound-related functionality. This includes both audio data (analog and digital) and the functionality that is used to directly control the audio environment, such as Volume and Tone Control. The Audio Device Class does not include functionality to operate transport mechanisms that are related to the reproduction of audio data, such as tape transport mechanisms or CD-ROM drive control.

1.2 PURPOSE

The purpose of this document is to describe the minimum capabilities and characteristics an audio device shall support to comply with the USB. This document also provides recommendations for optional features.

1.3 RELATED DOCUMENTS

- *Universal Serial Bus Specification*, Revision 2.0 (referred to in this document as the *USB Specification*). In particular, see Chapter 5, “USB Data Flow Model” and Chapter 9, “USB Device Framework.”
- *Universal Serial Bus 3.1 Specification*, Revision 1.0 (referred to in this document as the *USB 3.1 Specification*). This document covers details specific to SuperSpeed and SuperSpeed+ devices.
- *Universal Serial Bus Device Class Definition for Audio Data Formats* (referred to in this document as USB Audio Data Formats).
- *Universal Serial Bus Device Class Definition for Terminal Types* (referred to in this document as USB Audio Terminal Types).
- ANSI S1.11-1986 standard.
- MPEG-1 standard ISO/IEC 111172-3 1993.
- MPEG-2 standard ISO/IEC 13818-3 Feb. 20, 1997.
- Digital Audio Compression Standard (AC-3), ATSC A/52A Aug. 20, 2001. (available from http://www.atsc.org/)
- ANSI/IEEE-754 floating-point standard.
- ISO/IEC 60958 International Standard: *Digital Audio Interface and Annexes.*
- ISO/IEC 61937 standard.

1.4 TERMS AND ABBREVIATIONS

This section defines terms used throughout this document. For additional terms that pertain to the Universal Serial Bus, see Chapter 2, “Terms and Abbreviations,” in the *USB Specification*.

ASRC	Acronym for Asynchronous Sampling Rate Conversion.
(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 Cluster.
Audio Control Attribute	Parameter of an Audio Control. Examples are Current, Minimum, Maximum and Resolution attributes of a Volume Control.
Audio Control	Logical object that is used to manipulate a specific audio property. Examples are Volume Control, Mute Control, etc.
Audio data stream	Transport medium that can carry audio information.

AVA Documents Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

AEC	Acronym for Audio Echo Cancellation.
Audio Function	Independent part of a USB device that deals with audio-related functionality.
Audio Interface Association (AIA)	Grouping of a single AudioControl interface, zero or more AudioStreaming interfaces and zero or more MIDIStreaming interfaces that together constitute a complete interface to a particular version (compliance level) of the Audio Function.
AudioControl interface (ACI)	USB interface used to access the Audio Controls inside an Audio Function.
AudioStreaming interface (ASI)	USB interface used to transport audio streams into or out of the Audio Function.
Clock Multiplier (CM)	Multiplies a clock signal by a fractional multiplier.
Clock Selector (CX)	Selects from a number of input clock signals.
Clock Source (CS)	Generates an audio-related clock signal (master clock or sample clock).
CMD	Acronym for Clock Multiplier descriptor.
CSD	Acronym for Clock Source descriptor.
CXD	Acronym for Clock Selector descriptor.
Effect Unit (EU)	Provides advanced audio manipulation on the incoming logical audio channels.
Entity	Addressable logical object inside an Audio Function.
Extension Unit (XU)	Applies an undefined process to a number of logical input channels.
Feature Unit (FU)	Provides basic audio manipulation on the incoming logical audio channels.
FUD	Acronym for Feature Unit descriptor.
High Capability Descriptor	A new representation for class-specific descriptors that allows for potentially large (>256 bytes) and dynamic descriptors.
Input Pin	Logical input connection to an Entity. Carries a single Cluster.
Input Terminal (IT)	Receptacle for audio information flowing into the Audio Function.
ITD	Acronym for Input Terminal descriptor.
Logical Audio Channel	Logical transport medium for a single audio channel. Makes abstraction of the physical properties and formats of the connection. Is usually identified by spatial location. Examples are Left channel, Right Surround channel, etc.
MIDIStreaming interface (MSI)	USB interface that may be used to transport MIDI data streams into or out of the Audio Function.
Mixer Unit (MU)	Mixes a number of logical input channels into a number of logical output channels.

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

MUD	Acronym for Mixer Unit descriptor.
OTD	Acronym for Output Terminal descriptor.
Output Pin	Logical output connection to an Entity. Carries a single Cluster.
Output Terminal (OT)	An outlet for audio information flowing out of the Audio Function.
Phase Locked Loop	A hardware or software control system that generates an output signal whose phase is related to the phase of an input signal. Typically used in clock recovery applications.
Processing Unit (PU)	Applies a predefined process to a number of logical input channels.
PUD	Acronym for Processing Unit descriptor.
Reserved	Reserved is a keyword indicating reserved bits, bytes, words, fields, and code values that are set-aside for future standardization. Their use and interpretation may be specified by future extensions to this specification and, unless otherwise stated, shall not be utilized or adapted by vendor implementation. A reserved bit, byte, word, or field shall be set to zero by the sender and shall be ignored by the receiver.
RUD	Acronym for Sampling Rate Converter Unit descriptor.
Sampling Rate Converter Unit (RU)	Converts the incoming audio data stream, running at a sampling rate, synchronous to a first master clock, into an outgoing audio data stream that is running at a sampling rate, synchronous to a second master clock, which is free running with respect to the first master clock.
Selector Unit (SU)	Selects from a number of input Clusters.
SUD	Acronym for Selector Unit descriptor.
Terminal	Addressable logical object inside an Audio Function that represents a connection to the Audio Function’s outside world.
Unit	Addressable logical object inside an Audio Function that represents a certain audio sub-functionality.
XUD	Acronym for Extension Unit descriptor.

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

2 MANAGEMENT OVERVIEW

The USB is very well suited for transport of audio, ranging from low fidelity voice connections to high quality, multi-channel audio streams. The USB has become a ubiquitous connector on modern PC's and is well-understood by most consumers today. As such, it has become the connector of choice for many peripherals and is indeed the simplest and most pervasive digital audio connector available today. Consumers can count on this medium to meet all of their current and future audio needs. Many applications from communications, to entertainment, to music recording and playback, can take advantage of the audio features of the USB.

In principle, a versatile bus specification like the USB provides many ways to propagate and/or control digital audio. For the industry, however, it is very important that audio transport mechanisms be well defined and standardized on the USB. Only in this way can interoperability be guaranteed among the many possible audio devices on the USB. Standardized audio transport mechanisms also help to keep software drivers as generic as possible. The Audio Device Class described in this document satisfies those requirements. It is written and revised by experts in the audio field. Other device classes that address audio in some way should refer to this document for their audio interface specification.

An essential issue in audio is synchronization of the data streams. Indeed, the smallest artifacts are easily detected by the human ear. Therefore, a robust synchronization scheme on isochronous transfers has been developed and incorporated in the *USB Specification* and the *USB 3.1 Specification*. The Audio Device Class definition adheres to these synchronization schemes to transport audio data reliably over the bus.

This document contains all necessary information for a designer to build a USB-compliant device that incorporates Audio Functionality. It specifies the standard and class-specific descriptors that shall be present in each USB Audio Function. It further explains the use of class-specific requests that allow for full Audio Function control. A number of predefined data formats are listed and fully documented. Each format defines a standard way of transporting audio over the USB. Provisions have been made so that vendor-specific audio formats and compression schemes can be handled as well.

Many of the changes introduced in this version of the USB Specification for Audio Devices are inspired by the desire to use USB Audio in modern portable devices. Special attention has been paid to make the Audio Device Class more power-friendly by providing new tools to selectively enable and disable parts of the Audio Function and also by supporting burst mode data transfers for longer sleep times in between data transfers. In addition, the specification supports new codec types and data formats for consumer audio applications, provides numerous clarifications of the original specification and extensions to support various changes in the core specification.

2.1 OVERVIEW OF KEY DIFFERENCES BETWEEN ADC V2.0 AND V3.0

The following list is not an exhaustive list of all changes that have been introduced. For complete information, refer to the full specification.

- Dolby Processing Unit removed
- Encoder and Decoder support removed
- Copy protection (S/PDIF-style) removed
- bmAttributes (MaxPacketsOnly bit) field in class-specific isochronous endpoint descriptor removed
- Type II and Extended Type II Audio Data Formats removed
- New Power Domains
- New Multi-Function Processing Unit
- Additional Type III formats introduced
- New Cluster descriptor
- New Extended Terminal descriptor

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

- New Connectors descriptor
- Layout of all class-specific descriptors has changed
- New class-specific String descriptors
- New High Capability descriptors
- Additional sources for interrupts
- Provides for backward compatibility, using multiple Configuration descriptors
- Support for LPM/L1
- Mandated support for BADD on each ADC3.0 compliant device

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

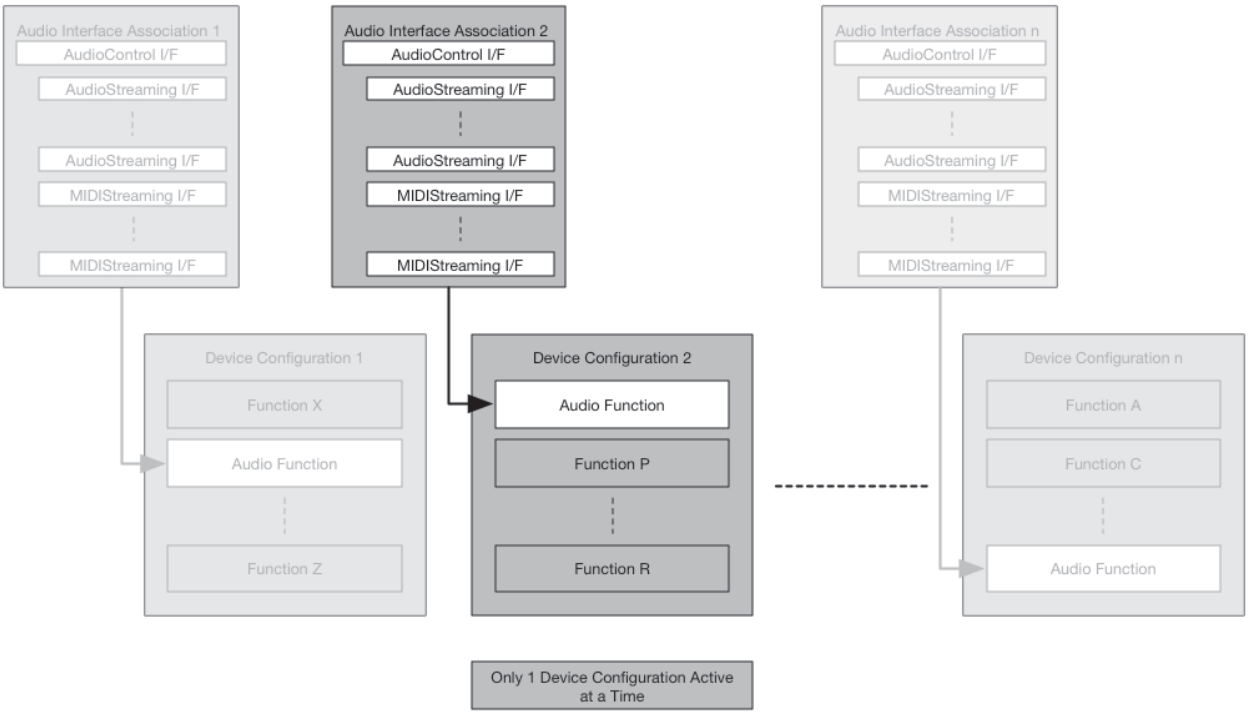
© USB-IF:1997-2016

3 FUNCTIONAL CHARACTERISTICS

3.1 INTRODUCTION

In many cases, Audio Functionality does not exist as a standalone device. It is one capability that, together with other functions, constitutes a “composite” device. A perfect example of this is a DVD-ROM player, which can incorporate video, audio, data storage, and transport control. The Audio Function is thus located at the interface level in the device class hierarchy. The following figure provides details.

Figure 3-1: Multiple Configurations and their Audio Interface Associations and Interfaces



An Audio Function is considered to be a ‘closed box’ that has very distinct and well defined interfaces to the outside world. Audio Functions are addressed and accessed through an Audio Interface Association (AIA). The AIA groups all USB interfaces that together provide access to the Audio Function for control and streaming purposes.

A USB Device may express different AIAs representing the same underlying hardware by using multiple USB Device Configuration descriptors. These AIAs may be used to provide access to the Audio Function at different compliance levels. For example, one AIA might provide access to the Audio Function in ADC 1.0 mode, whereas another AIA might provide access to the Audio Function in ADC 3.0 mode, as defined by this document. Note that the Audio Function may take on a different topology and even different functionality when accessed through different AIAs.

Because only one Configuration may be active at a time, only one AIA shall be active at a time for the same Audio Function. Host software should choose the most desirable Configuration and use the Standard SET_CONFIGURATION command to select the Configuration containing the most desired AIA or AIAs (if there are multiple independent Audio Functions residing in the same USB Device).

An AIA shall have a single AudioControl interface and can have zero or more AudioStreaming and zero or more MIDIStreaming interfaces. The AudioControl (AC) interface is used to access the Audio Controls of the function whereas the AudioStreaming (AS) interfaces are used to transport audio streams into and out of the Audio Function. MIDIStreaming (MS) interfaces can be used to transport MIDI data streams into and out of the Audio Function.

AVA Documents Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

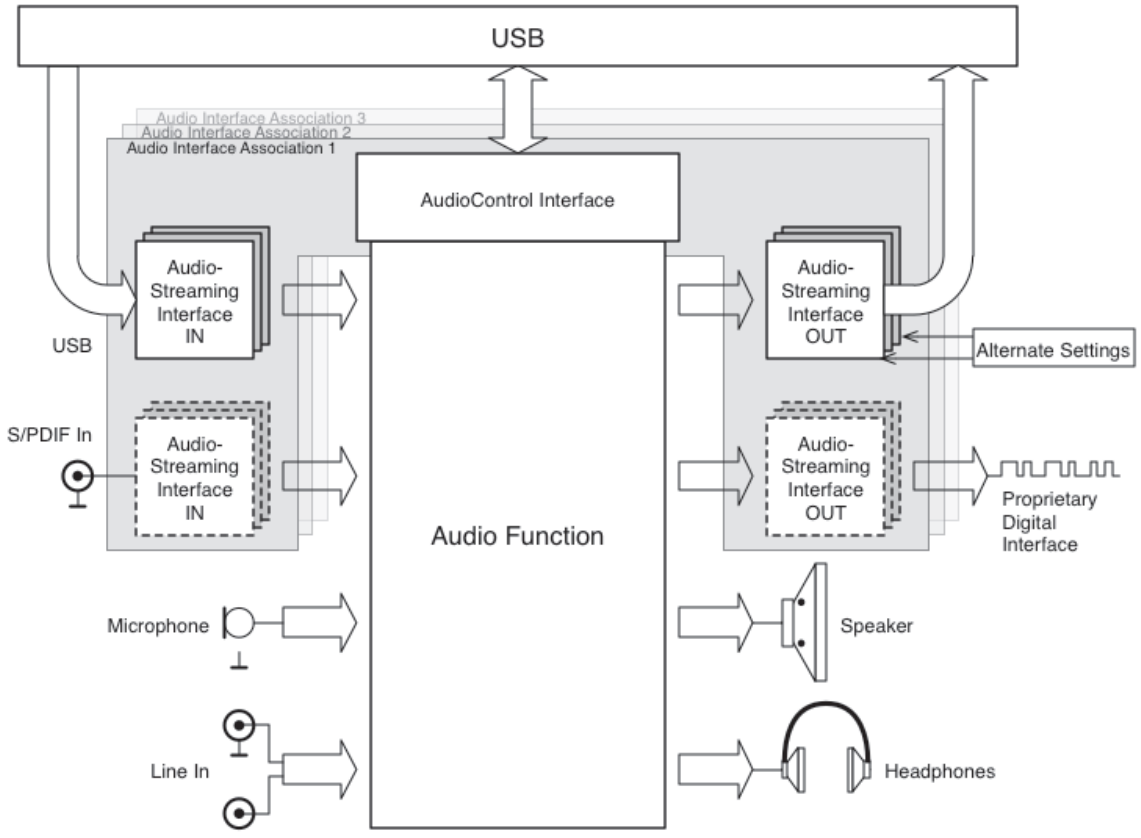
Note: All MIDI-related information is grouped in a separate document, *Universal Serial Bus Device Class Definition for MIDI Devices* that is considered part of this specification. The remainder of this document will therefore not mention MIDIStreaming interfaces and their specifics anymore.

Note that AudioStreaming interfaces do not necessarily all represent USB audio streams. Any audio stream, regardless of its origin or destination that enters or leaves the Audio Function may have an associated AudioStreaming interface, if that interface requires the Host to have access and/or visibility to some interface Controls or descriptors that may have an impact on the operation of the interface from a USB perspective.

A device can have *multiple independent* Audio Functions located in the same composite Device. They are each accessed through their active Audio Interface Association, which must be contained within the currently selected USB Configuration.

The following figure illustrates the concept of an Audio Function and its associated interfaces:

Figure 3-2: Audio Function Global View



All functionality pertaining to controlling parameters that directly influence audio perception (like volume) are located inside the central rectangle and are exclusively controlled through the AudioControl interface. Streaming aspects of the communication to or from the Audio Function are handled through separate AudioStreaming interfaces, as necessary. Each USB audio stream shall be represented by an AudioStreaming interface. A non-USB audio stream may be represented by an AudioStreaming interface if there is a need to convey information between that audio stream's interface and the Host. All control data that is related specifically to the streaming behavior of the interface is conveyed through the AudioStreaming interface.

A physical S/PDIF connection to the Audio Function is a typical example of a non-USB AudioStreaming interface. Although the actual audio data is coming in from the outside world (not through the USB), it might be necessary to control some aspects of the S/PDIF connection. In that case, the S/PDIF connection is represented by an AudioStreaming interface so that it becomes addressable through USB.

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

Also note that the connection between the AudioStreaming interfaces and the Audio Function is not 'solid'. The reason for this is that when seen from the inside of the Audio Function, each audio stream entering or leaving the Audio Function is represented by a special object, called a Terminal (see further). The Terminal concept abstracts the actual AudioStreaming interface inside the Audio Function and provides a logical view on the connection rather than a physical view. This abstraction allows audio channels within the Audio Function to be treated as 'logical' audio channels that do not have physical characteristics associated with them anymore (analog vs. digital, format, sampling rate, bit resolution, etc.).

3.2 BASIC AUDIO DEVICE DEFINITION

The Basic Audio Device Definition (BADD) exists as a separate document and is based on and fully compatible with this Audio Device Class Definition (ADC 3.0). As such, the BADD defines a subset of functionality commonly found in Headphones, Microphones, and Headsets using the features and tools provided by the ADC but it removes all of the optionality allowed by the ADC and prescribes rigorously how certain features shall be implemented. This allows simple Host drivers to know exactly what to expect when encountering a BADD-compliant device, reducing significantly the work a driver needs to perform parsing the BADD Device's descriptors and interacting with the Device.

3.3 BACKWARDS COMPATIBILITY

This 3.0 version of the *Audio Device Class Definition* (ADC 3.0) as such is not backwards compatible with any of the previous versions of the same specification.

However, to ensure the broadest possible interoperability among new ADC 3.0 Devices and legacy ADC 1.0 or ADC 2.0 Hosts, this specification requires that the ADC 3.0 compliant Device shall do the following:

- The first USB Configuration descriptor (index 0) exposed by the Device shall contain an Audio Interface Association (AIA), which is compliant with ADC 1.0 or ADC 2.0. In other words, if Host software selects the first indexed Device Configuration, it shall expose an Audio Function that is compliant with ADC 1.0 or ADC 2.0. This requirement should allow the Audio Function to interoperate with existing Hosts exactly the same as current legacy ADC 1.0 or ADC 2.0 Functions.
- If the Audio Function contains at least one USB Audio Streaming Interface, the Device shall have another Configuration descriptor that exposes an AIA which is BADD 3.0 compliant. A Device shall only have a single AIA of this type. However, this same BADD compliant AIA may need to be duplicated in more than one Configuration descriptor. This requirement guarantees that the Device will interoperate with new ADC 3.0 Hosts that support the BADD 3.0 specification. To activate this mode of operation for the Audio Function, the Host needs to select a Device Configuration which contains the BADD 3.0 compliant AIA.
- The Device may have one or more Configurations that expose one or more AIAs which shall be compliant with ADC 3.0. This way the Audio Function may provide functionality beyond what is available in BADD 3.0 operating mode. To activate this mode of operation, the Host needs to explicitly select a Configuration that contains the desired AIA(s).

It is strongly recommended that *ALL* ADC 3.0 Hosts support the BADD 3.0 specification.

3.4 AUDIO INTERFACE ASSOCIATION (AIA) AND INTERFACE ASSOCIATION DESCRIPTOR

On the USB, an Audio Function is completely defined by its interfaces. An Audio Function has one or more Audio Interface Associations (AIA) that provide access to it at different compliance levels. Only one AIA can be active at one time. Each AIA shall have one AudioControl interface and can have zero or more AudioStreaming interfaces.

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

The standard USB Interface Association mechanism is used to describe the Audio Interface Association i.e. to bind those interfaces together. Interface Association is expressed via the standard USB Interface Association descriptor (IAD). Every Interface Association descriptor has a **bFunctionClass**, **bFunctionSubClass** and **bFunctionProtocol** field that together identify the function that is represented by the Association. The following paragraphs define these fields for the Audio Device Class.

3.4.1 AUDIO FUNCTION CLASS

The Audio Function Class is contained in the **bFunctionClass** field of a Standard Interface Association descriptor. This specification requires that the Function Class code be the same as the Audio Interface Class code.

The Audio Function Class code is assigned by the USB-IF. For details, see Appendix A.1, “Audio Function Class Code”.

3.4.2 AUDIO FUNCTION SUBCLASS

The Audio Function Subclass is contained in the **bFunctionSubClass** field of a Standard Interface Association descriptor. Any AIA exposed by a Device that is compliant with this specification shall use this field to further define the device. This field indicates whether the AIA is a full-fledged implementation of the ADC 3.0 specification or is compliant with one of the BADD-defined Profiles.

The assigned codes can be found in A.2, “Audio Function Subclass Codes” of this specification. All other Subclass codes are unused and reserved by this specification for future use.

3.4.3 AUDIO FUNCTION PROTOCOL

The Audio Function Protocol is contained in the **bFunctionProtocol** field of a Standard Interface Association descriptor. The Function Protocol code is used to reflect the compliance level of the Audio Function so that enumeration software can decide which driver version needs to be instantiated. This specification requires that the Function Protocol code be the same as the Audio Interface (both AudioControl and AudioStreaming) Protocol code.

The assigned Protocol codes can be found in Appendix A.3, “Audio Function Protocol Codes” of this specification. All other Protocol codes are unused and reserved by this specification for future use.

3.5 AUDIO INTERFACE CLASS

The Audio Interface Class code is contained in the **bInterfaceClass** field of the Standard Interface descriptor of any Interface contained in an AIA. All USB Interfaces which provide functionality through this specification use the same Audio Interface Class code.

The Audio Interface class code is assigned by the USB. For details, see Appendix A.4, “Audio Interface Class Code”.

3.6 AUDIO INTERFACE SUBCLASS

The Audio Interface Subclass code is contained in the **bInterfaceSubClass** field of the Standard Interface descriptor of any Interface contained in an AIA. All USB interfaces that provide functionality through this specification, as well as any interfaces that provide MIDI streaming functionality, shall use one of the following Audio Interface Subclass codes as provided by this specification:

- AudioControl Interface Subclass
- AudioStreaming Interface Subclass
- MIDIStreaming Interface Subclass

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

The assigned codes can be found in Appendix A.5, “Audio Interface Subclass Codes” of this specification. All other Subclass codes are unused and reserved by this specification for future use.

3.7 AUDIO INTERFACE PROTOCOL

The Audio Interface Protocol code is contained in the **bInterfaceProtocol** field of the Standard Interface descriptor of any Interface contained in an AIA. The Interface Protocol code is used to reflect the compliance level of the Audio Function.

All Audio Functions compliant with this ADC 3.0 specification shall use Interface Protocol Code IP_VERSION_03_00. The assigned codes can be found in Appendix A.6, “Audio Interface Protocol Codes” of this specification. All other Protocol codes are unused and reserved by this specification for future use.

3.8 AUDIO FUNCTION CATEGORY

The Audio Function Category code is contained in the Class Specific Audio Control Interface descriptor as defined by this specification. This code indicates the primary intended use for the Audio Function. The following Function Categories are currently defined in this specification:

- Desktop Speaker: One or more speakers set up in a small environment to provide audio intended primarily for one person.
- Home Theater: Several speakers set up in a moderately sized environment to provide audio levels significantly louder than a Desktop Speaker setup and intended to be clearly heard by multiple people.
- Microphone: A device set up to record audio from audible sources.
- Headset: A device with at least one speaker and at least one microphone designed to be worn or held by a user to provide personal audio playback and voice input capabilities.
- Telephone: A Headset or handset type device that also connects to a telephone system, (e.g. POTs, PBX, VoIP) capable of making and receiving telephone calls.
- Converter: A device that allows conversion of audio from one electrical or optical format to another electrical or optical format, and/or converting audio data from one encoding format to another (e.g. AC-3 to PCM, etc.).
- Voice/Sound recorder: A device set up with at least one microphone and at least one speaker that is designed to operate, at least some of the time, independently of the Host to record and store audible sources and play back its recorded content.
- IO Box: A device designed to deliver one or more, possibly different, electrical and optical inputs and outputs for connection to other devices.
- Musical Instrument: A musical instrument, e.g. piano, guitar, synthesizer, drum machine, etc.
- Pro-Audio: A device not typically used by consumers of audio, e.g. editing equipment, multitrack recording equipment, etc.
- Audio/Video: The audio from a device that also supplies simultaneous video where the expectation is that the audio is tightly coupled to the video, e.g. a camcorder, a DVD player, a television, etc.
- Control Panel: A device that is used to control the flow of audio through a system of audio devices, such as a mixer panel.
- Other: Any device whose primary purpose is sufficiently different from the above descriptions as to be considered a completely different form of device.

The assigned codes can be found in Appendix A.7, “Audio Function Category Codes” of this specification. All other Category codes are unused and reserved by this specification for future use.

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

3.9 CLOCK DOMAINS

A Clock Domain is defined as a zone within which all sampling clocks are derived from the same master clock. Therefore, within the same Clock Domain, all sampling clocks are synchronous and their timing relationship is constant. However, the sampling clocks can be at different sampling frequencies. The master clock can be generated in many different ways. An internal crystal could be the master clock; the USB start of frame (SOF) could be used or even an externally supplied clock could serve as a master clock.

Multiple different Clock Domains may exist within the same Audio Function.

3.10 POWER DOMAINS

A Power Domain is defined as a zone within the Audio Function that groups one or more Entities and allows the Host to control power consumption levels for that Domain. This way, the Host can potentially switch parts of the Audio Function to lower power states when these parts are currently not in use, leading to an overall decrease in power consumption. As an example, a USB headset may have two separate Power Domains, one for the output-related functionality (headphone) and one for the input-related functionality (microphones). When the headset is used to simply listen to music, the input-related functionality may be temporarily switched to a low power state (or even completely off) to conserve power. This is especially important if the headset is used in conjunction with a battery-powered device, such as a mobile phone or MP3 player, where extending battery life is important.

Multiple different Power Domains may exist within the same Audio Function. Power Domains are identified by a unique Power Domain ID within the Audio Function.

3.11 AUDIO SYNCHRONIZATION TYPES

Each isochronous audio endpoint used in an AudioStreaming interface belongs to a synchronization type as defined in Section 5 of the *USB Specification*. The following sections briefly describe the possible synchronization types.

3.11.1 ASYNCHRONOUS

Asynchronous isochronous audio endpoints produce or consume data at a rate that is locked either to a clock external to the USB or to a free-running internal clock. These endpoints are not synchronized to a start of frame (SOF) or to any other clock in the USB domain.

3.11.2 SYNCHRONOUS

The clock system of synchronous isochronous audio endpoints can be controlled externally through SOF or Bus Interval synchronization. Such an endpoint shall lock its sample clock to the SOF tick or to the start of a new Bus Interval.

3.11.3 ADAPTIVE

Adaptive isochronous audio endpoints are able to source or sink data at any rate within their operating range. This implies that these endpoints shall run an internal process that allows them to match their natural data rate to the data rate that is imposed at their interface.

3.11.4 IMPLICATIONS OF THE DIFFERENT SYNCHRONIZATION TYPES

The following sections provide some information on the impact the different synchronization types have on both the Device and Host implementation. Several scenarios are considered and for each synchronization type, the implications on Host and Device side are listed.

AVA Documents Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

3.11.4.1 SINGLE DIRECTION SINK ENDPOINT

A typical example for this scenario is a USB speaker, implementing a sink endpoint that receives streaming audio data from the Host.

- Asynchronous
  - **Host**: The Host driver needs to be able to handle an explicit feedback endpoint. From the feedback data, the Host then decides how many samples to send over the data streaming endpoint in subsequent Bus Intervals.
  - **Device**: The Device has its own local, free-running audio sample clock, which determines how many samples are consumed by the Device each Bus Interval. The Device shall implement an explicit feedback endpoint as well as the necessary logic to provide the correct feedback values to send over said endpoint back to the Host. The advantage of this mode of operation is that it is fairly easy to generate a robust, stable, jitter-free, high-quality audio sample clock (derived from a crystal-based master clock, for example).
- Synchronous
  - **Host**: The Host needs to send out a known number of bytes for each packet going to the Device. The Host may need to generate a (fixed) pattern of audio samples to achieve the desired sampling rate. As an example, to generate a sampling rate of 44.1 kHz in a Full-Speed implementation, the Host needs to send a repeating pattern of nine packets containing 44 audio samples, followed by one packet containing 45 audio samples.
  - **Device**: This synchronization type requires the Device to implement either an audio clock PLL or an ASRC.
- Adaptive
  - **Host**: The Host may use any method or means to determine how many samples per Bus Interval to transmit. Effectively operating as a “Synchronous-to-SOF” Source is an easy approach, but not the only one allowed by the USB specification.
  - **Device**: This synchronization type requires the Device to implement either an audio clock PLL or an ASRC to adapt to the average number of samples arriving over a certain period of time.

3.11.4.2 SINGLE DIRECTION SOURCE ENDPOINT

A typical example for this scenario is a USB microphone, implementing a source endpoint that sends streaming audio data to the Host.

- Asynchronous
  - **Host**: The Host needs to operate as an Adaptive Sink. Depending on the Host’s system design, this may require an ASRC or an audio clock PLL on the Host side.
  - **Device:** The Device has its own local, free-running audio sample clock, which determines how many samples are produced by the Device each Bus Interval. The advantage of this mode of operation is that it is fairly easy to generate a robust, stable, jitter-free, high-quality audio sample clock (derived from a crystal-based master clock, for example).
- Synchronous
  - **Host**: The Host receives a known number of audio samples in each packet coming from the Device. The Host should expect a fixed pattern of audio samples to achieve the desired sampling rate.

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

- **Device**: This synchronization type requires the Device to implement either an audio clock PLL to lock on to the SOF or the start of Bus Interval and generate a high-quality audio sample clock directly or use an ASRC as a bridge between the local and USB clock domains. The Device shall generate a fixed packet size pattern as described in the *USB Audio Data Formats Definition* document.

- Adaptive
  - **Host:** This scenario requires the implementation of a feedforward OUT endpoint in the Device that allows the Host to inform the Device how many samples per Bus Interval to send to the Host over the streaming data IN endpoint.
  - **Device:** This synchronization type requires the Device to implement an audio clock PLL or an ASRC to adapt to the sample rate being communicated by the Host via the feedforward OUT endpoint.

3.11.4.3 SOURCE AND SINK ENDPOINTS

A typical example for this scenario is a USB headset, implementing both a sink endpoint that receives streaming audio data from the Host and a source endpoint that sends streaming audio data to the Host.

In general, the endpoints can be treated independently, using the same rules and guidelines as stated above. This includes the ability for the Device to support separate clocks on each endpoint.

However, if the Device has its own clock and both data streams share this clock, the following two special cases can be used to allow implicit feedback:

- Both endpoints are Asynchronous. In this case, the data rate that appears on the Source endpoint can be used by the Host as implicit feedback to adjust the data rate transmitted to the Sink endpoint.
- Both endpoints are Adaptive. In this case, the data rate that appears on the Sink endpoint can be used by the Device to adjust the data rate transmitted by the Source endpoint.

Note: Using either of these two implicit feedback mechanisms would preclude the ability of an ADC 3.0 compliant device from being able to use Power Domains to shut down either the source or the sink, as data must remain flowing in both directions to provide the feedback/feedforward information.

3.12 INTER CHANNEL SYNCHRONIZATION

An important issue when dealing with audio, and 3-D audio in particular, is the phase relationship between different physical audio channels. Indeed, the virtual spatial position of an audio source is directly related to and influenced by the phase differences that are applied to the different physical audio channels used to reproduce the audio source. Therefore, it is imperative that USB Audio Functions respect the phase relationship among all related audio channels. However, the responsibility for maintaining the phase relation is shared among the USB host software, hardware, and all of the audio peripheral devices or functions.

3.13 AUDIO FUNCTION TOPOLOGY

To be able to manipulate the physical properties of an Audio Function, its functionality shall be divided into addressable Entities. Two types of such generic Entities are identified and are called Units and Terminals. In addition, a special type of Entity is defined. These Entities are called Clock Entities and they are used to describe and manipulate the clock signals inside the Audio Function.

Units provide the basic building blocks to fully describe most Audio Functions. Audio Functions are built by connecting together several of these Units. A Unit has one or more Input Pins and a single Output Pin, where each Pin represents a Cluster of logical audio channels inside the Audio Function (see Section 3.13.1, "Cluster").

AVA Documents Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

Units are wired together by connecting their I/O Pins according to the required topology. Note that it is perfectly legal to connect the Output Pin of an Entity to multiple Input Pins residing on different other Entities, effectively creating a one-to-many connection.

In addition, the concept of a Terminal is introduced. There are two types of Terminals. An Input Terminal (IT) is an Entity that represents a starting point for audio channels inside the Audio Function. An Output Terminal (OT) represents an ending point for audio channels inside the Audio Function. From the Audio Function's perspective, a USB endpoint is a typical example of an Input or Output Terminal. It either provides data streams to the Audio Function (IT) or consumes data streams coming from the Audio Function (OT). Likewise, a Digital to Analog converter, built into the Audio Function is represented as an Output Terminal in the Audio Function's model. Connection to the Terminal is made through its single Input or Output Pin.

Input Pins of a Unit are numbered starting from one up to the total number of Input Pins on the Unit. The Output Pin number is always one. Input Terminals have only one Output Pin and its number is always one. Output Terminals have only one Input Pin and it is always numbered one.

The information, traveling over I/O Pins is not necessarily of a digital nature. It is perfectly possible to use the Unit model to describe fully analog or even hybrid Audio Functions. The mere fact that I/O Pins are connected together is a guarantee (by construction) that the protocol and format, used over these connections (analog or digital), is compatible on both ends.

Every Unit in the Audio Function is fully described by its associated Unit descriptor (UD). The Unit descriptor contains all necessary fields to identify and describe the Unit. Likewise, there is a Terminal descriptor (TD) for every Terminal in the Audio Function. In addition, these descriptors provide all necessary information about the topology of the Audio Function. They fully describe how Terminals and Units are interconnected.

This specification describes the following types of standard Units and Terminals that are considered adequate to represent most Audio Functions:

- Input Terminal (IT)
- Output Terminal (OT)
- Mixer Unit (MU)
- Selector Unit (SU)
- Feature Unit (FU)
- Sampling Rate Converter Unit (RU)
- Effect Unit (EU)
- Processing Unit (PU)
- Extension Unit (XU)

Besides Units and Terminals, the concept of a Clock Entity is introduced. Three types of Clock Entities are defined by this specification:

- Clock Source (CS)
- Clock Selector (CX)
- Clock Multiplier (CM)

A Clock Source provides a certain sampling clock frequency to all or part of the Audio Function. A Clock Source can represent an internal sampling frequency generator, but it can also represent an external sampling clock signal input to the Audio Function.

A Clock Source has a single Clock Output Pin that carries the sampling clock signal, represented by the Clock Source. The Clock Output Pin number is always one.

A Clock Selector is used to select between multiple sampling clock signals that might be available inside an Audio Function. It has multiple Clock Input Pins and a single Clock Output pin. Clock Input Pins are numbered

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

starting from one up to the total number of Clock Input Pins on the Clock Selector. The Clock Output Pin number is always one.

A Clock Multiplier is used to derive a new clock signal with a different frequency from the clock signal at its single Clock Input Pin. It does this by multiplying that clock signal frequency by a numerator P and then dividing it by a denominator Q. The new clock signal is guaranteed to be synchronous with the input clock signal. A Clock Multiplier has one Input Pin and one Output Pin and their numbers are always one. An implementation can choose to make the values P and Q programmable, although most Audio Functions will expose P and Q Controls as Read-Only and report any changes to those values via the interrupt mechanism. Such changes may be necessary when the Audio Function detects a change in native sampling rate for incoming encoded content, for example.

By using a combination of Clock Source, Clock Selector, and Clock Multiplier Entities, the most complex clock systems can be represented and exposed to Host software.

Clock Input and Output Pins are fundamentally different from Input and Output Pins defined for Units and Terminals. Clock Pins carry only clock signals and therefore cannot be connected to Unit or Terminal Input and Output Pins. They are only used to express clock circuitry topology.

Each Input and Output Terminal has a single Clock Input Pin that is connected to a Clock Output Pin of a Clock Entity. The clock signal carried by that Clock Output Pin determines at which sampling frequency the hardware represented by the Terminal is operating.

Each Sampling Rate Converter Unit has two Clock Input Pins that are typically connected to the Clock Output Pins of two different Clock Entities. The clock signals carried by those Clock Output Pins determine the sampling frequencies between which the Sampling Rate Converter Unit is converting.

Each Clock Entity is described by a Clock Entity descriptor (CED). The Clock Entity descriptor contains all necessary fields to identify and describe the Clock Entity.

The descriptors are further detailed in Section 4, “Descriptors” of this document.

The ensemble of Unit descriptors, Terminal descriptors and Clock Entity descriptors provide a full description of the Audio Function to the Host. This information is typically retrieved from the device at enumeration time. By parsing the descriptors, a generic audio driver should be able to fully control the Audio Function, except for the functionality represented by Extension Units. Those require vendor-specific extensions to the audio class driver.

Important Note:

> The complete set of Audio Function descriptors provides only a static initial description of the Audio Function. During operation, a number of events can happen that force the Audio Function to change its state. Host software shall be notified of these changes to remain ‘in sync’ with the Audio Function at all times. An extensive interrupt mechanism is in place to report any and all state changes to Host software.

Figure 3-3, “Inside the Audio Function” illustrates the concepts defined above. Using the iconic symbols defined further, it describes a hypothetical Audio Function that incorporates 15 Entities: three Input Terminals, five Units, three Output Terminals, two Clock Sources, a Clock Selector, and one Clock Multiplier. Each Entity has its unique ID (from 1 to 15) and descriptor that fully describes the functionality of the Entity and also how that particular Entity is connected into the overall topology of the Audio Function.

Input Terminal 1 (IT 1) is the representation of a USB OUT endpoint used to stream audio from the Host to the audio device. IT 2 is the representation of an analog Line-In connector on the audio device whereas IT 3 is an analog Microphone-In connector on the audio device. Selector Unit 4 (SU 4) selects between the audio coming from the Host and the audio present at the Line-In connector. Feature Unit 5 (FU 5) is then used to manipulate

AVA Documents Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

the audio (Volume, Bass, Treble …) before it is presented to Output Terminal 9 (OT 9). OT 9 is the representation of a Headphone Out jack on the audio device.

At the same time, all three input sources (USB OUT, Line-In, and Mic-In) are connected to a Mixer Unit 6 (MU 6) that effectively mixes the three sources together. The output of the Mixer is then fed into a Processing Unit 7 (PU 7) that performs some audio processing algorithm(s) on the mix. The result is in turn sent to FU 8 where some final adjustments to the audio (Volume …) are made. FU 8 is connected to OT 10 and OT 11. OT 10 represents speakers incorporated into the audio device and OT 11 represents a USB IN endpoint used to send the processed audio to the Host for recording purposes.

Clock Source 12 (CS 12) represents an internal sampling frequency generator, running at 96 kHz for instance. Clock Source 15 (CS 15) is the representation of an external master sampling clock input that can be used to synchronize the device to an external source. Clock Selector 13 (CS 13) enables selection between the two available Clock Sources. The output of CS 13 provides a sampling frequency to IT 1, IT 2, IT3, OT 10, and OT 11 of 96 kHz. Clock Multiplier CM 14 further multiplies that clock signal by 0.5, providing a sampling frequency of 48 kHz to OT 9 for driving the headphone. Since all sampling frequencies used inside the Audio Function are at all times derived from a single master clock (internal or external), all audio streams in the Audio Function are synchronous.

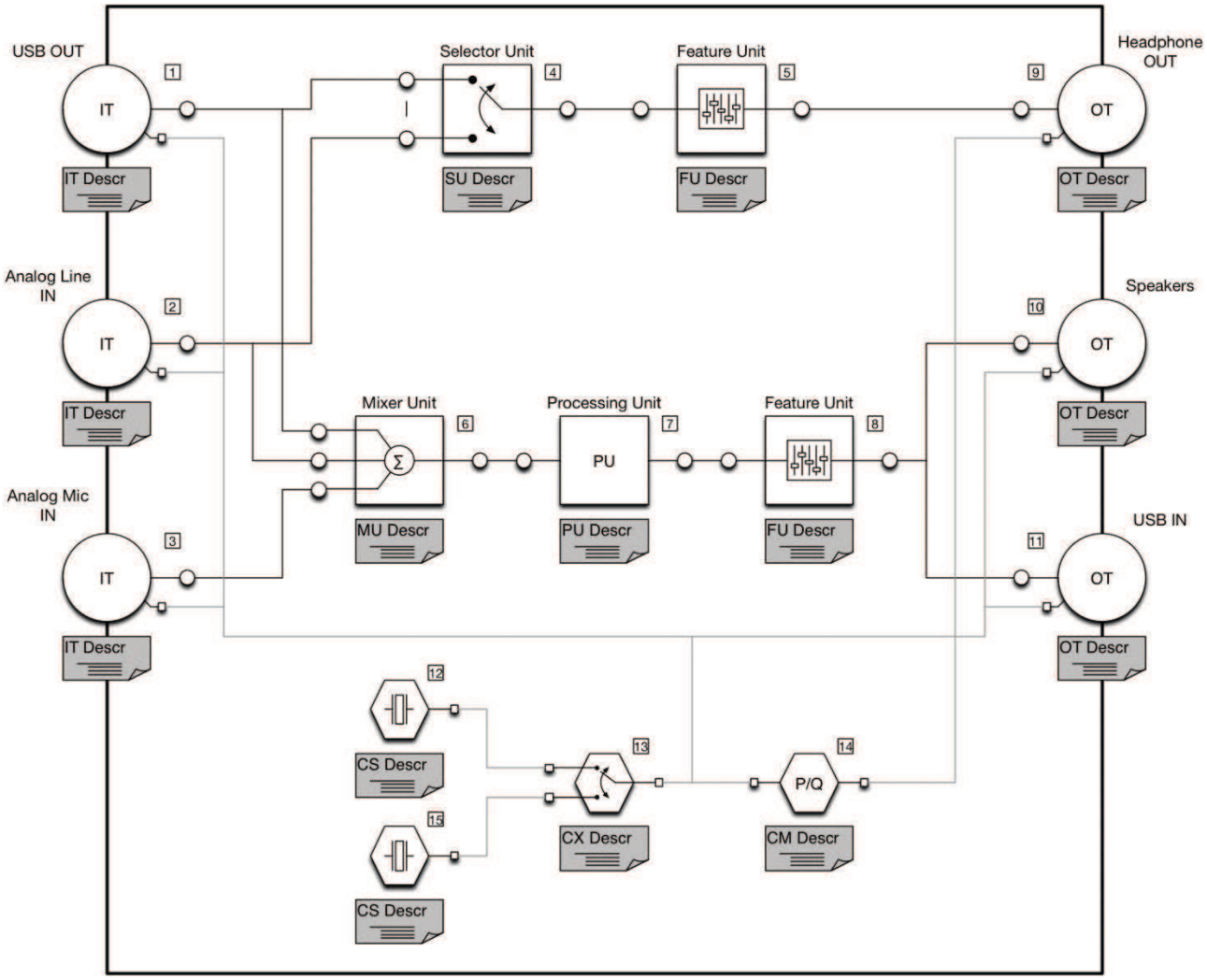
The descriptors, associated with each Entity clearly indicate to the Host what the exact nature of each Entity is. For instance, the IT 2 descriptor contains a field that indicates to the Host that it represents an external connector on the device, used as an analog Line In. Likewise, the MU 6 descriptor has a field that indicates that its Input Pin 1 is connected to the Output Pin of IT 1, Input Pin 2 is connected to the Output Pin of IT 2, and Input Pin 3 is connected to the Output Pin of IT 3.

For further details on descriptor contents, refer to Section 4, “Descriptors” of this document.

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

Figure 3-3: Inside the Audio Function



Inside an Entity, functionality is further described through Audio Controls. A Control typically provides access to a specific audio or clock property. Each Control has a set of attributes that can be manipulated or that present additional information on the behavior of the Control. A Control can have the following attributes:

- Current setting attribute
- Range attribute triplet consisting of:
  - Minimum setting attribute
  - Maximum setting attribute
  - Resolution attribute

As an example, consider a Volume Control inside a Feature Unit. By issuing the appropriate Get requests, the Host software can obtain values for the Volume Control's attributes and, for instance, use them to correctly display the Control on the screen. Setting the Volume Control's current attribute allows the Host software to change the volume setting of the Volume Control.

Additionally, each Entity in an Audio Function can have a memory space attribute. This attribute optionally provides generic access to the internal memory space of the Entity. This could be used to implement vendor-specific control of an Entity through generically provided access.

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

3.13.1 CLUSTER

A Cluster is a grouping of audio channels that carry tightly related synchronous audio information. Inside the Audio Function, complete abstraction is made of the actual physical representation form of the audio data that travels over the connections among Terminals and Units. Each audio channel in the Cluster is considered to be a logical channel and all the physical attributes of the channel (bit width, bit resolution, etc.) are not specified and considered irrelevant in the context of the Audio Function as seen through the AudioControl interface. The fact that an Input Pin and an Output Pin are connected together in the Audio Function's topology is a guarantee (by construction) that the information flowing over the connection is compatible with both Entities that are connected.

Channel numbering in the Cluster starts with channel one up to the number of channels in the Cluster. The virtual channel zero is used to address a master Control in a Unit (if present), effectively influencing all the channels at once. Note that the master Control (if present) shall be implemented separate from the per-channel Controls. Changing the setting of a master Control does not affect the settings of any of the individual channel Controls. The maximum number of independent channels in a Cluster is limited to 255 (Channel zero is used to reference the master channel).

A Cluster is characterized by a number of attributes:

- A unique identifier for the Cluster
- The number of audio channels in the Cluster
- The purpose of each audio channel, such as Voice, Speech, etc.
- The audio channel relationship (or spatial location) of each audio channel in the Cluster
- An indication whether an audio channel is part of an Ambisonic group

There is a Cluster descriptor (CD) associated with each Cluster that fully describes the Cluster.

There are two types of Clusters used in this specification:

- A logical Cluster describes audio channels within the Audio Function (the closed box) where audio channels are treated as logical concepts.
- A physical Cluster describes physical audio channels, for example when travelling over a connector.

3.13.2 INPUT TERMINAL

The Input Terminal (IT) is used to interface between the Audio Function's 'outside world' and other Units in the Audio Function. It serves as a receptacle for audio information flowing into the Audio Function. Its function is to represent a source of incoming audio data after this data has been properly extracted from the original audio stream into the separate logical channels that are embedded in this stream (the decoding process). The logical channels are grouped into a Cluster and leave the Input Terminal through a single Output Pin.

An Input Terminal can represent inputs to the Audio Function other than USB OUT endpoints. A Line-In connector on an audio device is an example of such a non-USB input. However, if the audio stream is entering the Audio Function by means of a USB OUT endpoint, there is a one-to-one relationship between the AudioStreaming interface that contains this endpoint and its associated Input Terminal. The class-specific interface descriptor contains a field that holds a direct reference to this Input Terminal. The Host needs to use both the AudioStreaming interface and endpoint descriptors and the Input Terminal descriptor to get a full understanding of the characteristics and capabilities of the Input Terminal. Stream-related parameters are stored in the AudioStreaming interface descriptors. Control-related parameters are stored in the Terminal descriptor.

The conversion process from incoming, possibly encoded, audio streams to logical audio channels always involves some kind of decoding engine. The decoding types range from rather trivial decoding schemes like converting interleaved stereo 16 bit PCM data into a Left and Right logical channel to very sophisticated

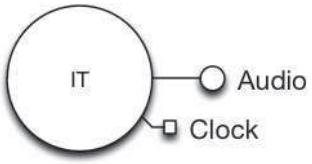
AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

schemes like converting an MPEG-2 7.1 encoded audio stream into Front Left, Front Left of Center, Front Center, Front Right of Center, Front Right, Back Left, Back Right and Low Frequency Effects logical channels. The decoding engine is considered part of the Entity that actually receives the encoded audio data streams (like a USB AudioStreaming interface). The type of decoding is therefore implied by the value in the **bmFormats** field, located in the class-specific AudioStreaming interface descriptor. The associated Input Terminal deals with the logical channels after they have been decoded.

An Input Terminal has a single Clock Input Pin. The clock signal present at that Pin is used as the sampling clock for all underlying hardware that is represented by this Input Terminal. There is a field in the Input Terminal descriptor that uniquely identifies the Clock Entity to which the Input Terminal is connected.

The symbol for the Input Terminal is depicted in the following figure:

Figure 3-4: Input Terminal Icon



3.13.3 OUTPUT TERMINAL

The Output Terminal (OT) is used to interface between Units inside the Audio Function and the 'outside world'. It serves as an outlet for audio information, flowing out of the Audio Function. Its function is to represent a sink of outgoing audio data before this data is properly packed from the original separate logical channels into the outgoing audio stream (the encoding process). The Cluster enters the Output Terminal through a single Input Pin.

An Output Terminal can represent outputs from the Audio Function other than USB IN endpoints. A speaker built into an audio device or a Line Out connector is an example of such a non-USB output. However, if the audio stream is leaving the Audio Function by means of a USB IN endpoint, there is a one-to-one relationship between the AudioStreaming interface that contains this endpoint and its associated Output Terminal. The class-specific interface descriptor contains a field that holds a direct reference to this Output Terminal. The Host needs to use both the AudioStreaming interface and endpoint descriptors and the Output Terminal descriptor to fully understand the characteristics and capabilities of the Output Terminal. Stream-related parameters are stored in the AudioStreaming interface descriptors. Control-related parameters are stored in the Terminal descriptor.

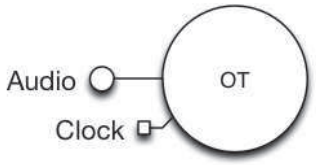
The conversion process from incoming logical audio channels to possibly encoded audio streams always involves some kind of encoding engine. The encoding engine is considered part of the Entity that actually transmits the encoded audio data streams (like a USB AudioStreaming interface). The type of encoding is therefore implied by the value in the **bmFormats** field, located in the class-specific AudioStreaming interface descriptor. The associated Output Terminal deals with the logical channels before encoding.

An Output Terminal has a single Clock Input Pin. The clock signal present at that Pin is used as the sampling clock for all underlying hardware that is represented by this Output Terminal. There is a field in the Output Terminal descriptor that uniquely identifies the Clock Entity to which the Output Terminal is connected.

The symbol for the Output Terminal is depicted in the following figure:

AVA Documents Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

1014 **Figure 3-5: Output Terminal Icon**



1015

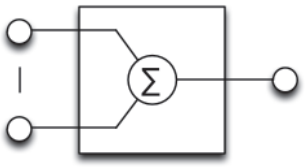
1016 ### 3.13.4 MIXER UNIT

1017 The Mixer Unit (MU) transforms a number of logical input channels into a number of logical output channels.
1018 The input channels are grouped into one or more Clusters. Each Cluster enters the Mixer Unit through an Input
1019 Pin. The logical output channels are grouped into one Cluster and leave the Mixer Unit through a single Output
1020 Pin.

1021 Every input channel can virtually be mixed into all of the output channels. If n is the total number of input
1022 channels and m is the number of output channels, then there is a two-dimensional array (n · m) of Mixer
1023 Controls in the Mixer Unit. Not all of these Controls have to be physically implemented. Some Controls can
1024 have a fixed setting and be non-programmable. The Mixer Unit descriptor reports which Controls are
1025 programmable in the **bmControls** bitmap field. Using this model, a permanent connection can be implemented
1026 by reporting the Control as non-programmable in the **bmControls** bitmap and by returning a Control setting of
1027 0 dB when requested. Likewise, a missing (muting) connection can be implemented by reporting the Control as
1028 non-programmable in the **bmControls** bitmap and by returning a Control setting of –∞ dB. A Mixer Unit SHALL
1029 respond to the appropriate Get Request to allow the Host to retrieve the actual settings on each of the (n · m)
1030 Mixer Controls.

1031 The symbol for the Mixer Unit can be found in the following figure:

1032 **Figure 3-6: Mixer Unit Icon**



1033

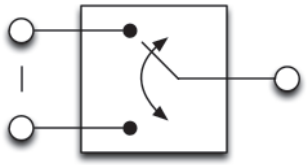
1034 ### 3.13.5 SELECTOR UNIT

1035 The Selector Unit (SU) selects from n Clusters, each containing m logical input channels and routes them
1036 unaltered to the single output Cluster, containing m output channels. It represents a multi-channel source
1037 selector, capable of selecting between n m-channel sources. It has n Input Pins and a single Output Pin.

1038 The symbol for the Selector Unit can be found in the following figure:

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

Figure 3-7: Selector Unit Icon



3.13.6 FEATURE UNIT

The Feature Unit (FU) is essentially a multi-channel processing unit that provides basic manipulation of multiple single-parameter Audio Controls on the incoming logical channels. For each logical channel, the Feature Unit optionally provides Audio Controls for the following features:

- Mute
- Volume
- Tone Control (Bass, Mid, Treble)
- Graphic Equalizer
- Automatic Gain Control
- Delay
- Bass Boost
- Loudness
- Input Gain
- Input Gain Pad
- Phase Inverter

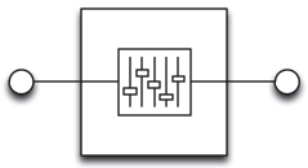
In addition, the Feature Unit optionally provides the above Audio Controls but now influencing all channels of the Cluster at once. In this way, 'master' Controls can be implemented. The master Controls are cascaded after the individual channel Controls. This setup is especially useful in multi-channel systems where the individual channel Controls can be used for channel balancing and the master Controls can be used for overall settings.

The logical channels in the Cluster are numbered from one to the total number of channels in the Cluster. The 'master' channel has channel number zero and is always virtually present.

The Feature Unit descriptor reports what Controls are present for every channel in the Feature Unit and for the 'master' channel. All logical channels in a Feature Unit are fully independent. There exist no cross couplings among channels within the Feature Unit. There are as many logical output channels, as there are input channels. These are grouped into one Cluster that enters the Feature Unit through a single Input Pin and leaves the Unit through a single Output Pin.

The symbol for the Feature Unit is depicted in the following figure:

Figure 3-8: Feature Unit Icon



3.13.7 SAMPLING RATE CONVERTER UNIT

The Sampling Rate Converter (SRC) Unit (RU) is included here as an optional way to indicate where exactly within the Audio Function sampling rate conversion takes place. In many cases, it is unnecessary to indicate this

AVA Documents
Document Sharing Hub

point and any SRC Unit can be omitted from the topology without materially affecting the information presented to the Host. The primary reason to include the SRC Unit is to accurately report any latencies incurred by the Sampling Rate Conversion process.

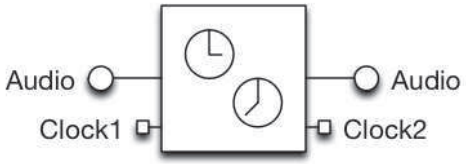
The Sampling Rate Converter Unit provides a bridge function between different clock domains within the Audio Function. The Sampling Rate Converter does not provide any Audio Controls. It takes the audio on all the logical channels in the single input Cluster belonging to a certain clock domain and converts them into the same logical channels in the single output Cluster but now belonging to another clock domain.

There are as many logical output channels, as there are input channels. These are grouped into one Cluster that enters the SRC Unit through a single Input Pin and leaves the Unit through a single Output Pin.

A SRC Unit has two Clock Input Pins. One Clock Input Pin is associated with the single Input Pin of the SRC Unit. The other Clock Input Pin is associated with the single Output Pin of the SRC Unit. The clock signals present at those two Clock Input Pins identify the two clock domains between which the SRC Unit is converting. Note that it is allowed to have both Clock Input Pins connected to clock signals belonging to the same clock domain.

The symbol for the SRC Unit is depicted in the following figure:

Figure 3-9: Sampling Rate Converter Unit Icon



3.13.8 EFFECT UNIT

The Effect Unit (EU) is a multi-channel processing unit that provides advanced manipulation of a multi-parameter Audio Control on the incoming logical channels on a per-channel basis. For each logical channel, the Effect Unit provides one of the following Audio Controls:

- Parametric Equalizer Section
- Reverberation
- Modulation Delay
- Dynamic Range Compressor

In addition, the Effect Unit optionally provides one of the above Audio Controls but now influencing all channels of the Cluster at once. In this way, a ‘master’ Control can be implemented. The master Control is cascaded after the individual channel Controls. This setup is especially useful in multi-channel systems where the individual channel Controls can be used for channel balancing and the master Control can be used for overall settings.

The logical channels in the Cluster are numbered from one to the total number of channels in the Cluster. The ‘master’ channel has channel number zero and is always virtually present.

The Effect Unit descriptor reports what Controls are present for every channel in the Effect Unit and for the ‘master’ channel. All logical channels in an Effect Unit are fully independent. There exist no cross couplings among channels within the Effect Unit. There are as many logical output channels, as there are input channels. These are grouped into one Cluster that enters the Effect Unit through a single Input Pin and leaves the Unit through a single Output Pin.

If the Effect Unit is an explicit member of a Power Domain, then switching the Power Domain to any Power State other than D0 shall render the Unit non-functional and its output is undefined. Explicit bypass functionality may be required to preserve the integrity of the signal path.

AVA Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

© USB-IF:1997-2016

3.13.8.1 PARAMETRIC EQUALIZER SECTION EFFECT UNIT

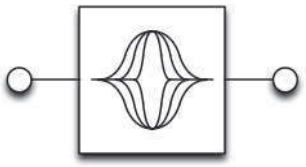
The Parametric Equalizer Section (PEQS) Effect Unit is used to manipulate and equalize the frequency characteristics of the original audio information around a certain center frequency. In order to build a parametric equalizer, a number of these PEQS Effect Units may need to be cascaded to obtain the desired functionality. The parameters that can be manipulated to obtain the desired equalizing effect are:

- Center Frequency: the frequency around which the audio spectrum is manipulated. Expressed in Hz.
- Q Factor: a measure for the range of frequencies around the center frequency that are influenced. Expressed as a ratio.
- Gain: the amount of gain or attenuation at the center frequency. Expressed in dB.

The algorithm to produce the desired equalization effect can be manipulated on a per-channel basis. The master channel concept allows equalization for all channels simultaneously.

The symbol for the PEQS Processing Unit can be found in the following figure:

Figure 3-10: PEQS Effect Unit Icon



3.13.8.2 REVERBERATION EFFECT UNIT

The Reverberation Effect Unit is used to add room acoustics effects to the original audio information. These effects can range from small room reverberation effects to simulation of a large concert hall reverberation. A number of parameters can be manipulated to obtain the desired reverberation effects.

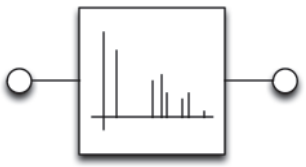
- Reverb Type: Room1, Room2, Room3, Hall1, Hall2, Plate, Delay, and Panning Delay.
- Reverb Level: sets the amount of reverberant sound versus the original sound. Expressed as a ratio.
- Reverb Time: sets the time over which the reverberation will continue. Expressed in s.
- Reverb Delay Feedback: used with Reverb Types Delay and Delay Panning. Sets the way in which delay repeats. Expressed as a ratio.
- Reverb Pre-Delay: sets the delay time between original sound and initial reverb reflection. Expressed in ms.
- Reverb Hi-Freq Roll-Off: sets the cut-off frequency of a low pass filter on the reflections. Expressed in Hz.

It is entirely left to the designer how a certain reverberation effect is obtained. It is not the intention of this specification to precisely define all the parameters that influence the reverberation experience (for instance in a multi-channel system, it is possible to create very similar reverberation impressions, using different algorithms and parameter settings on all channels).

The symbol for the Reverberation Effect Unit can be found in the following figure:

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01

1143 **Figure 3-11: Reverberation Effect Unit Icon**



1144

1145 ### 3.13.8.3 MODULATION DELAY EFFECT UNIT

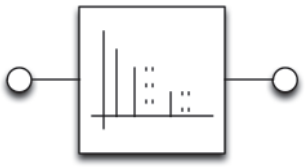
1146 The Modulation Delay Effect Unit is used to add modulation (like chorus) effects to the original audio
1147 information. A number of parameters can be manipulated to obtain the desired modulation effects.

1148 • Modulation Delay Balance: controls the ratio of the original sound to that of the effected sound. Expressed
1149 as a ratio.
1150 • Modulation Delay Rate: sets the speed (frequency) of the modulator. Expressed in Hz.
1151 • Modulation Delay Depth: sets the depth at which the sound is modulated. Expressed in ms.
1152 • Modulation Delay Time: sets the delay that is added to the modulated sound before adding it to the
1153 original sound. Expressed in ms.
1154 • Modulation Delay Feedback Level: controls the amount of the modulated sound that is routed back to the
1155 input of the modulator unit. Expressed as a ratio.

1156 It is entirely left to the designer how a certain modulation effect is obtained.

1157 The symbol for the Modulation Delay Effect Unit can be found in the following figure:

1158 **Figure 3-12: Modulation Delay Effect Unit Icon**



1159

1160 ### 3.13.8.4 DYNAMIC RANGE COMPRESSOR EFFECT UNIT

1161 The Dynamic Range Compressor Effect Unit is used to intelligently limit the dynamic range of the original audio
1162 information. A number of parameters can be manipulated to influence the desired compression.

Documents
Document Sharing Hub
Full document details: E DIN EN IEC 62680-1-5:2020-01