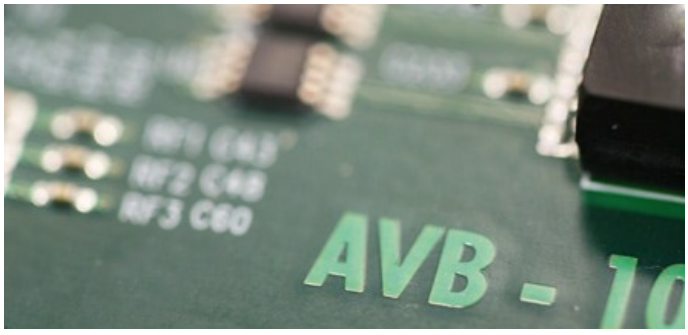
RIEDEL

AVB-Manager

User Manual



CONTENT

1 Preface 4

1.1 Version ... 5
1.2 About AVB ... 6

2 Installation 7

3 AVB-Manager 10

3.1 Main-Window ... 11
3.1.1 Main Menu ... 13
3.1.2 Settings Dialog ... 14
3.1.3 AVB Manager Network Info ... 15
3.1.4 Network Adapters ... 15
3.1.5 View-Selector ... 15
3.1.6 Logging-View ... 16
3.1.7 Log Files ... 17
3.2 Views ... 18
3.2.1 Entity properties ... 18
3.2.2 Connection Management ... 43
3.2.3 Firmware Update ... 49

4 How to ... 51

4.1 setup/delete connections ... 51
4.2 select a configuration ... 52
4.3 setup clocking ... 53
4.4 update the firmware ... 54

5 Appendix 57

5.1 Glossary ... 57
5.2 Service ... 58

1 Preface

Thank you for choosing a Riedel product.

This document provides detailed information about the installation and operation of the Riedel AVB-Manager. This Operating Manual is dedicated to engineers and field technicians.

NOTICE
This manual, as well as the software and any examples contained herein are provided “as is” and are subject to change without notice. The content of this manual is for informational purpose only and should not be construed as a commitment by Riedel Communications GmbH & Co. KG or its suppliers. Riedel Communications GmbH & Co. KG gives no warranty of any kind with regard to this manual or the software, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Riedel Communications GmbH & Co. KG shall not be liable for any errors, inaccuracies or for incidental or consequential damages in connection with the furnishing, performance or use of this manual, the software or the examples herein. Riedel Communications GmbH & Co. KG reserves all patent, proprietary design, title and intellectual property rights contained herein, including, but not limited to, any images, text, photographs incorporated into the manual or software.

AVB-Manager User Manual 1.2

1.1 Version

	This manual refers to AVB-Manager Version 1.10.x.x

Version Check

- The version is displayed in the startup splash screen and in the title bar of the AVB-Manager window.
- An information window can be opened by choosing the menu **Help** ▶ **About**.

Figure 1: About dialog

1.2 About AVB

Audio Video Bridging (AVB), also known as Ethernet AVB, is the common name for the set of technical standards developed by the IEEE 802.1 Audio Video Bridging Task Group. Ethernet AVB adds latency guarantees and bandwidth reservation for media streams to the existing Ethernet protocols. Ethernet AVB requires AVB-compatible Ethernet hardware (switches & NICs) but remains backward compatible with existing Ethernet standards. This means that only the sections of a network that are supposed to send or receive AVB streams need to be updated to AVB compatible hardware. Unlike IEEE 1588 PTP based standards, AVB includes a stream reservation protocol, allowing one to use the same network infrastructure for IP-services (e.g. fileserver access, corporate network) and communications without risking delays or a loss of audio.

AVB Standards

Ethernet AVB is built upon a basis of three major standards:

- 802.1Qav specifies queuing and forwarding rules that shape the traffic to avoid bottlenecks at any bridge or end station.
- P802.1Qat: defines the stream reservation protocol that sets up the path allowing a stream to go across the AVB cloud.
- P802.1AS allows microsecond accurate time synchronization across all AVB nodes.

Media is packetized following the IEEE P1733 and IEEE P1722 standards.

	The AVB communication protocol works on layer 2. Hence, it is not necessary to open any port in the firewall on the operating PC.

	The AVB commands are based on the AVB standards and only working if supported by the entity.

2 Installation

This chapter describes the installation procedure of the AVB-Manager.

The installation of the AVB-Manager includes also the mandatory installation of :

- Microsoft .NET Framework 4.0.30319
- WinPcap 4.1.3

The installation of these products will be skipped if they are already installed on the target PC.

➢ Start the installation by executing the Installer.exe file

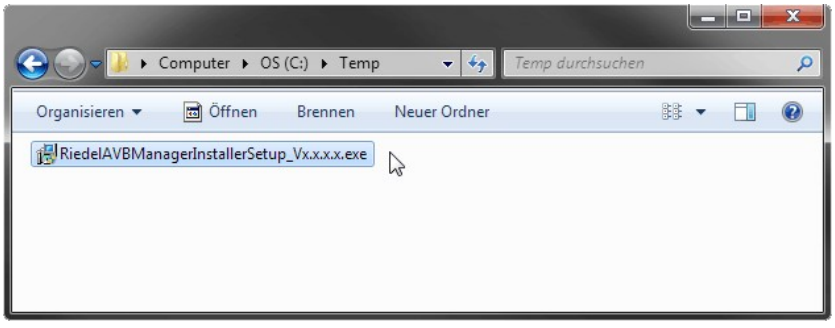


Figure 2: Installation File

The mandatory installations are displayed in the next window.
Already installed packages are unchecked and tagged with the information (Installed).

➢ Click *Install* to proceed

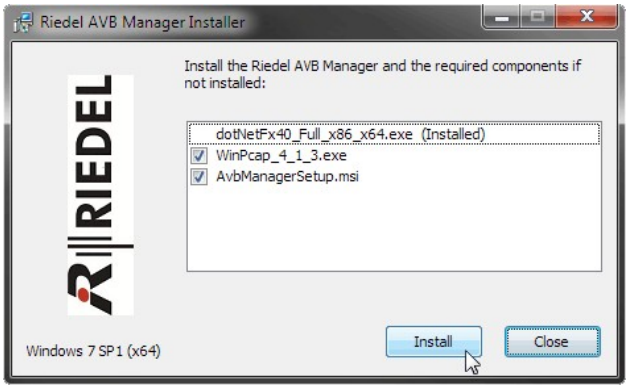


Figure 3: AVB Manager Installer

After installing Microsoft .NET Framework and WinPcap (if required) the installation of the AVB-Manager starts.

➢ Click *Next* to proceed

Figure 4: AVB Manager Setup

The destination folder can be defined in the next window.

➢ Click *Next* to proceed

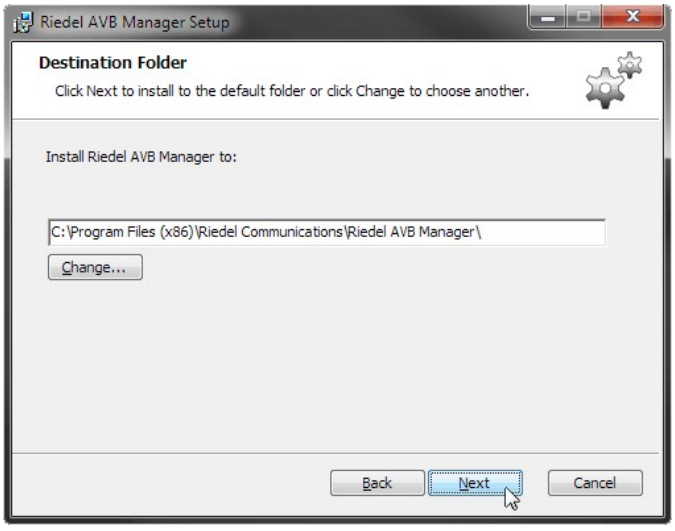


Figure 5: AVB Manager Setup - Destination Folder

Now the installer is ready to install the AVB-Manager.

➢ Click *Install* to proceed

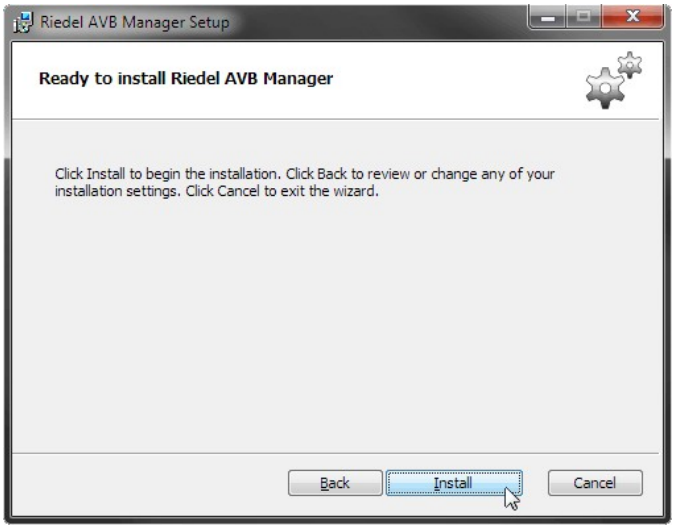


Figure 6: AVB Manager Setup - Ready to Install

➢ Clicking *Finish* exits the installer

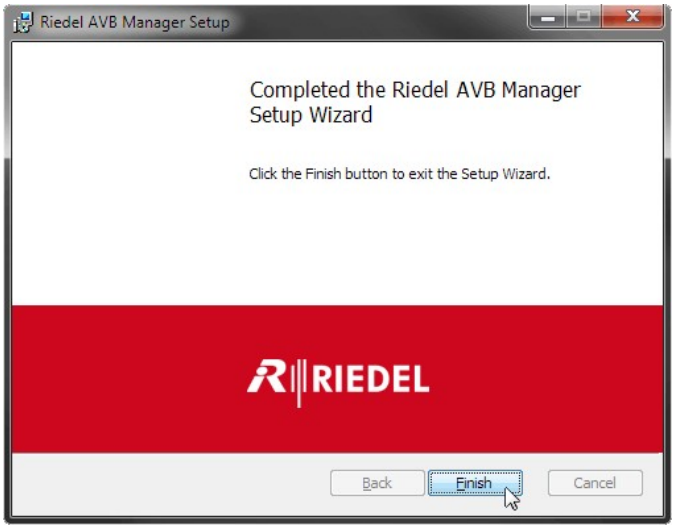


Figure 7: AVB Manager Setup - Installation completed

Another windows informs about the successful installation of all packages.

➢ Clicking *OK* to close the installer

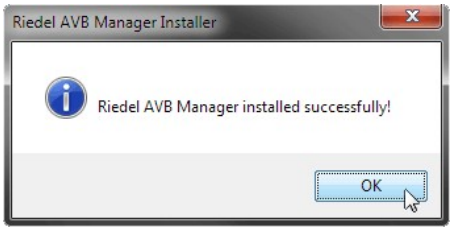


Figure 8: AVB Manager Installer - Successful Installation

3 AVB-Manager

This chapter describes the operation of the AVB-Manager.

The AVB-Manager can be started by double clicking the desktop icon.

Figure 9: Desktop Icon of the AVB-Manager

Multiple Instances

It is possible to start for each network adapter a separate instance of the AVB-Manager.
The entity id of an instance is generated from the network adapter mac address.
An instance set a mutex object to protect the generated entity ID against other instances.
The user settings are stored in the registry to save the settings per instance.

Command Line Option

The AVB-Manager can also be started with command line options.

The command line option /? or /help displays an overview about all supported commands.

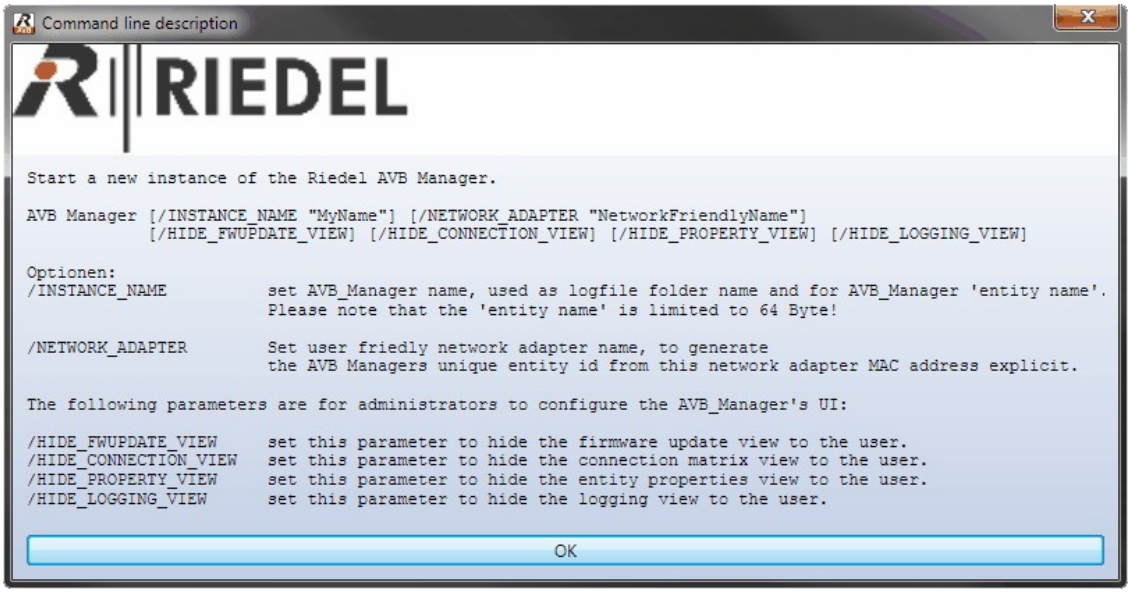


Figure 10: Command line description

3.1 Main-Window

The main window contains 5 areas:

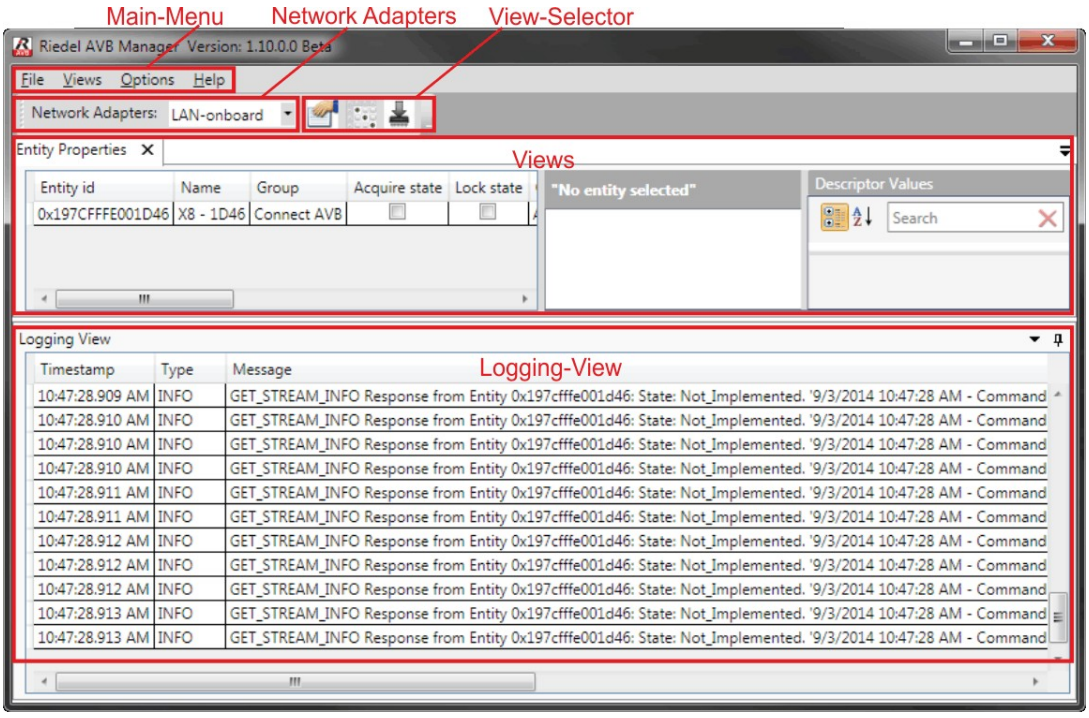


Figure 11: AVB-Manager main window (click to jump to the respective chapter)

Move the Docking Windows

The docking windows can be moved by drag'n'drop.
Click and hold the desired headline and move it to the new position - also outside the AVB-Manager window.
The arrows on the top, bottom, left and right side as well as in the middle in the AVB-Manager window allows the docking of the window to the respective position. While hovering over the arrows, a blue shadow shows the new area of the docking window.

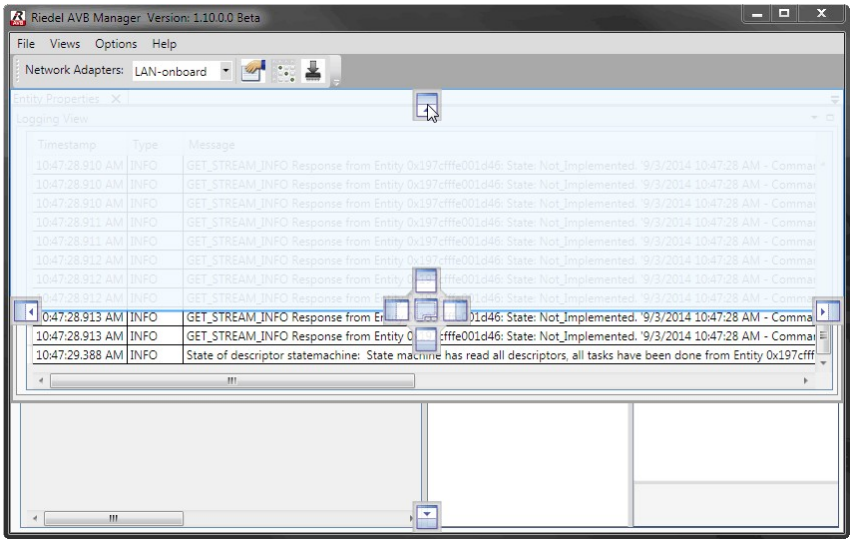


Figure 12: Move Docking Window

The docking windows can also be stacked with other docking windows onto a staple. To do so, drag'n'drop the docking window onto the middle icon in the center. In this case a docking windows can be displayed by clicking on the respective tab below.

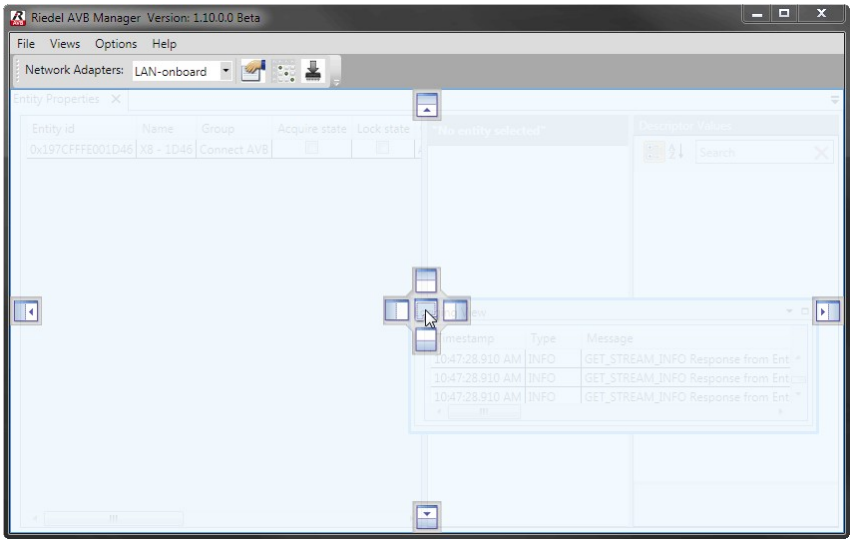


Figure 13: Staple Docking Window

3.1.1 Main Menu

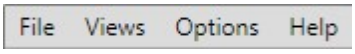
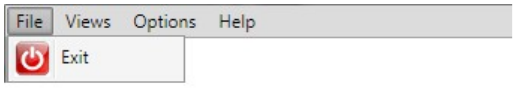
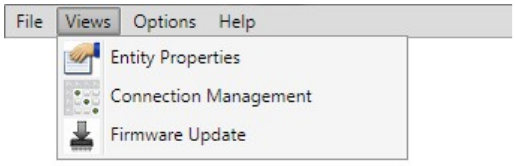
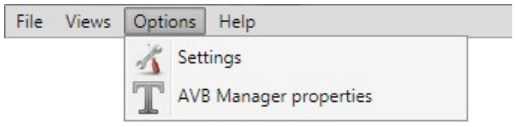
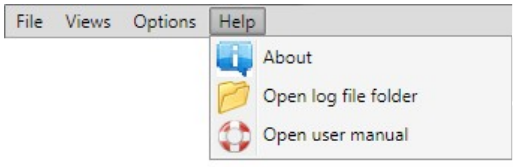


Figure 14: Main Menu

	**Exit**	Closes the AVB-Manager application.
	**Entity Properties**	Open the Entity Properties View in the Topic-Content area.
	Connection Management	Open the Connection Management View in the Topic-Content area.
	Firmware Update	Open the Firmware Update View in the Topic-Content area.
	**Settings**	Displays the Settings dialog.
	AVB Manager properties	Displays the windows to enter the entity name of the AVB Manager.
	**About**	Displays the About dialog
	Open log file folder	Opens the folder of the AVB-Manager log files in the windows explorer.
	Open user manual	Opens the user manual in windows-help format.

3.1.2 Settings Dialog

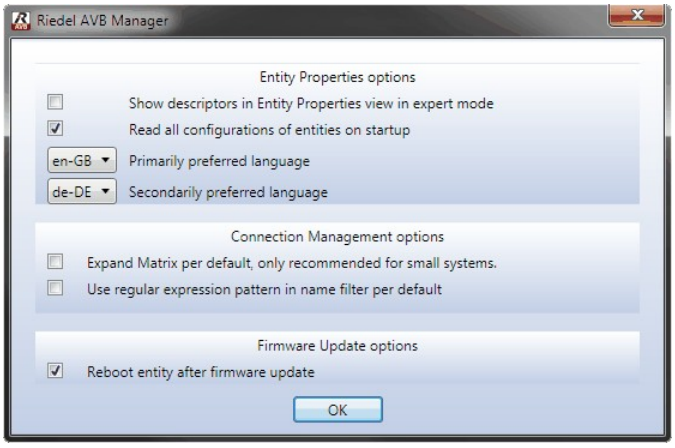


Figure 15: Settings dialog

Entity Property options (active after restarting the AVB Manger)	**Show descriptors in Entity Properties view in expert mode** By default only the major Descriptor Values are displayed. Enabling this checkbox displays all available Descriptor Values in the respective Descriptor groups.	
	Read all configurations of an entity on startup By default all configurations stored in an entity will be read on startup of the AVB Manager. This box can be unchecked to reduce network traffic and to speed up startup delay. In this case only the active configuration will be read during startup of the AVB Manager.	
	Primarily preferred language	Changes the language of the 'Locale' descriptor, if supported by the entity.
	Secondarily preferred language	
Connection Matrix options (active after re-opening the Connection Management tab)	**Expand Matrix per default, only recommended for small systems** By default the Connection Management opens with collapsed device Matrix. This box can be checked to expand the view and show all available streams. This mode requires more system performance during startup. Hence, this function should only activated for small systems.	
	Use regular expression pattern in name filter per default If checked, the checkboxes in the Connection Management properties are checked by default.	
Firmware Update options (only working if supported by the entity)	**Reboot entity after Firmware Update per default**	

3.1.3 AVB Manager Network Info

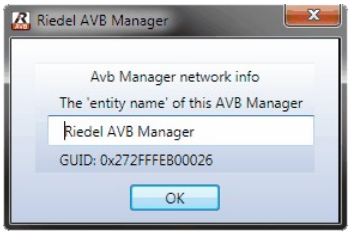


Figure 16: AVB Manager Network Info

In this windows the entity name of the AVB Manager is entered.
Furthermore the Global Unique Identifier of the AVB Manager is displayed.

3.1.4 Network Adapters

Figure 17: Network Adapters

In this field the network adapter needs to be selected, which is connected to the AVB network.
After changing the network adapter, all entities will be (re)discovered.

3.1.5 View-Selector

In this area the active View can be selected.

Figure 18: View-Selector

The selected Views are displayed in separate tabs in the Views area.
The Views can be switched by clicking on the respective tab.

Figure 19: Views-Tabs (click to jump to the respective chapter)

3.1.6 Logging-View

System messages and occurrences can be found in the logging view.

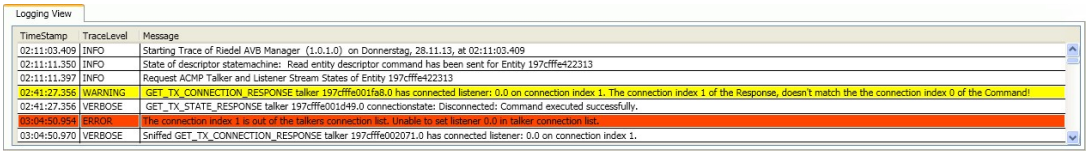


Figure 20: Logging View

Timestamp	Timestamp of the entry in the format hh:mm:ss.fff (seconds with 3 fractional digits)	
TraceLevel	INFO	Short Information Messages
	VERBOSE	Detailed Information Messages
	WARNING	Minor Warning Messages
	ERROR	Major Error Messages
	EXCEPTION	Debug information
Message	Text message	

Descending / ascending sorting of the messages is possible by clicking in head row of the respective column.

Only the last 2500 messages are displayed in the Logging-View.
Previous messages can be examined in the log files.

3.1.7 Log Files

The AVB-Manager stores every system message in a log file.
The log files of each instance are stored in a separate temp-folder of your system and can easily opened by using the main menu *Help -> open log file folder*.

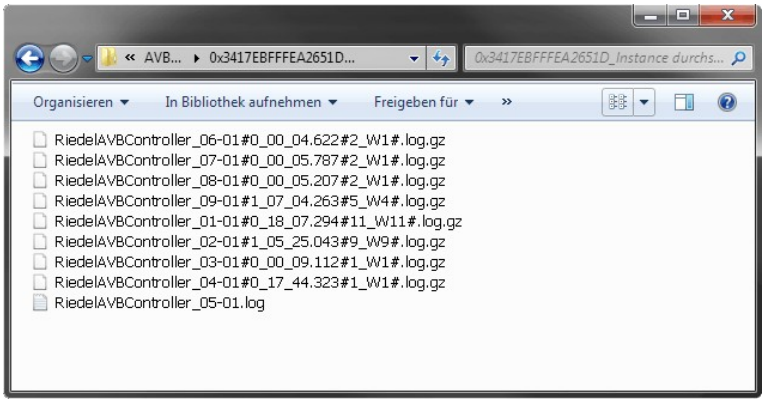


Figure 21: Log File Folder

The AVB-Manager stores the last ten sessions in separate files. The eleventh session overwrites the first one.
The maximum file size of a single log file is up to 100 MB and maximum 10% of free disk space.
Log files of greater size are splitted into multiple files.
To reduce disk space the files will be compressed (.gz) after closing the AVB Manager session.

The syntax of the filename is as follows:

%temp%\Riedel Communications\AVBManager**Multi-instance-Mutex**_Instance\ RiedelAVBController_**XX**-**YY**#**Timestamp**#**Counter**_***Wn_En_Xn***#.log.gz	
Multi instance Mutex	entity id of the AVB-Manager
XX	file number (01 ... 10 / round robin)
YY	increment of splitted files (01, 02, 03, ...)
Timestamp	relative timestamp of file termination
Counter	total number of all messages
_En	number of Errors (*if existing*)
_Wn	number of Warnings (*if existing*)

The log of a session starts in the first increment file with the information about Stack, GUI version and the start time:

```
Starting Trace of Riedel AVDECC Controller Protocol Stack Release_v1.0.0 at 2014-04-1 3:38:25.557 PM
Starting Trace of Riedel AVB Manager Version 1.6.3.0 (Build:Release) at 2014-04-1 3:38:25.635 PM
```

After this the different messages are listed, f.i. which network interface is connected to the AVB network or if new devices are detected or disconnected. The message `Trace stopped` is the last entry in the log file and is written by closing the AVB-Manager session. So it is possible to see if the log file is complete.

3.2 Views

The single contents of each view are described in the next sub chapters:

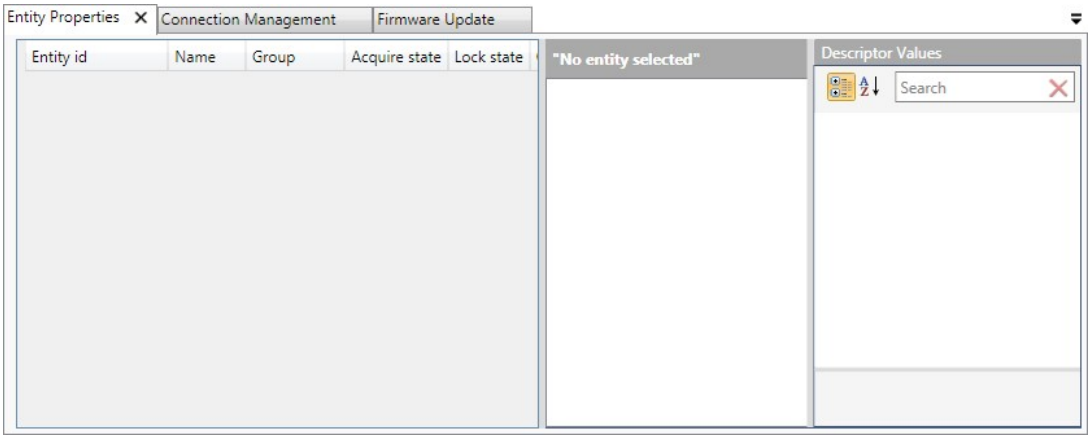


Figure 22: Views

3.2.1 Entity properties

The entity properties tab contains 3 areas:

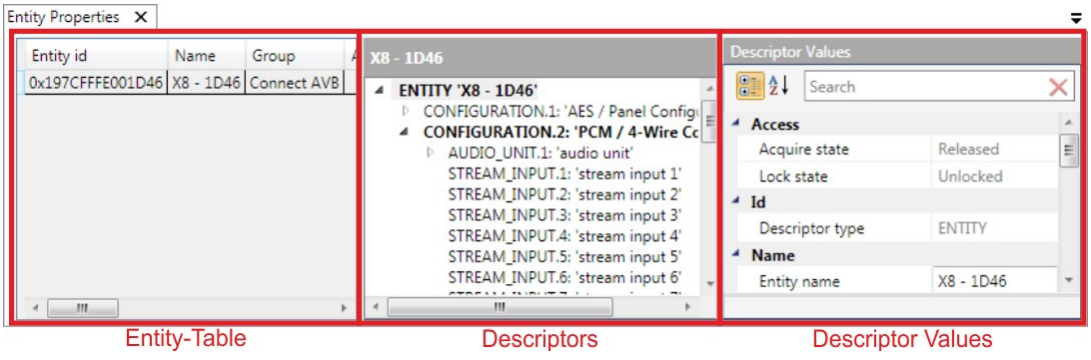


Figure 23: Entity properties (click to jump to the respective chapter)

3.2.1.1 Entity-Table

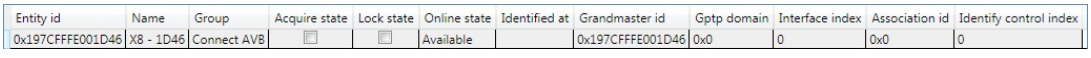


Figure 24: Entity-Table

The table of entities contains 12 columns:

Column	Content
Entity id	Global Unique Identifier of the entity.
Name	Identify the entity be user name.
Group	Identify multiple entities in a user defined group.
Acquire state	Check this box to obtain exclusive access to an entire entity or a sub-tree of objects. It also functions as a long-term lock on the entire entity or a sub-tree of objects. Unchecked this box to release the entity.

Lock state	Check this box to provide short-term exclusive access to the entity to perform atomic operations. When an entity is locked, it only accepts commands which alter state from the AVDECC Controller which locked the entity. Unchecked this box to unlock the entity.	
Online state	State of the entity	
	Available	Entity is available in the AVB network
	Departing	Entity departed in the AVB network and is no longer available
	Timeout	Entity doesn't send alive signal any more and is no longer available
Identified at	Timestamp of last recent identification notification, received from the entity	
Grandmaster id	This field is used to differentiate between devices in different gPTP domains. The value of gptp_grandmaster_id is per interface. If the GPTP_SUPPORTED flag is set in entity_capabilities then the gptp_grandmaster_id field is set to the Clock Identity of the grandmaster in the gPTP domain that this Entity is participating in on the interface transmitting the ADPDU. If the GPTP_SUPPORTED flag is not set in entity_capabilities then the gptp_grandmaster_id field is set to zero (0) on transmit and ignored on receive.	
Gptp domain	This field is used to differentiate between devices in different gPTP domains. The value of gptp_domain_number is per interface. If the GPTP_SUPPORTED flag is set in entity_capabilities then the gptp_domain_number field is set to the domainNumber of the grandmaster in the gPTP domain that this Entity is participating in on the interface transmitting the ADPDU. If the GPTP_SUPPORTED flag is not set in entity_capabilities then the gptp_domain_number field is set to zero (0) on transmit and ignored on receive.	
Interface index	This field is used to advertise the AEM AVB_INTERFACE descriptor that represents the interface that is transmitting this ADPDU. If the AEM_INTERFACE_INDEX_VALID flag is set in entity_capabilities then the interface_index field is set to the AVB_INTERFACE descriptor index of the interface transmitting the ADPDU. If the AEM_INTERFACE_INDEX_VALID flag is not set in entity_capabilities then the interface_index field is set to zero (0) on transmit and ignored on receive.	
Association id	This field is used to associate multiple AVDECC Entities into a logical collection. This allows each loudspeaker of a multi-channel rig to be a separate AVDECC Entity but to be associated by the AVDECC Controller into a single logical AVDECC Entity. If the ASSOCIATION_ID_SUPPORTED flag is not set in the entity_capabilities field then this field is set to zero (0). If ASSOCIATION_ID_SUPPORTED flag is set and ASSOCIATION_ID_VALID flag is not set in the entity_capabilities field then this field is set to zero (0). If ASSOCIATION_ID_SUPPORTED and ASSOCIATION_ID_VALID flags are set in the entity_capabilities field then the value of this field is the EUI-64 used to associate the AVDECC Entities (all associated AVDECC Entities will have the same EUI-64 in this field).	
Identify control index	This field is used to advertise the appropriate AEM CONTROL descriptor that is the primary IDENTIFY Control for the AVDECC Entity. If the AEM_IDENTIFY_CONTROL_INDEX_VALID flag is set in entity_capabilities then the identify_control_index field is set to a CONTROL descriptor index that can be used with the SET_CONTROL AEM command. If the AEM_IDENTIFY_CONTROL_INDEX_VALID flag is not set in entity_capabilities then the identify_control_index field is set to zero (0) on transmit and ignored on receive.	

One entity can be selected by left-click in the respective row.
The selected row will be highlighted and the available descriptors are displayed.
It is not possible to select multiple rows at the same time.

A popup menu appears by right-click on an entity:

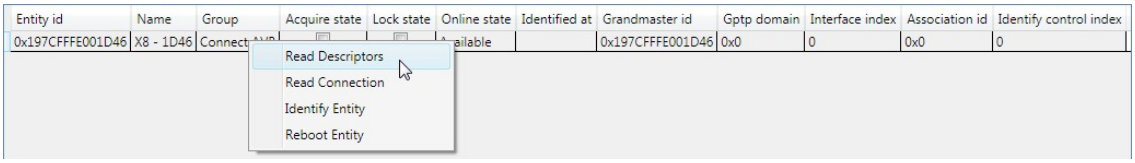


Figure 25: Right-click on Entity-Table

Command	Function
Read Descriptors	All Descriptor Values of the selected entity will be read.
Read Connection	Talker and Listener Connection States of the selected entity will be read.
Identify Entity	The respective entity signalize itself to be identified by the operator. The signalisation is device dependent, i.e. all LEDs will flash for a few seconds.
Reboot Entity	The selected entity will be rebooted.

3.2.1.2 Descriptors

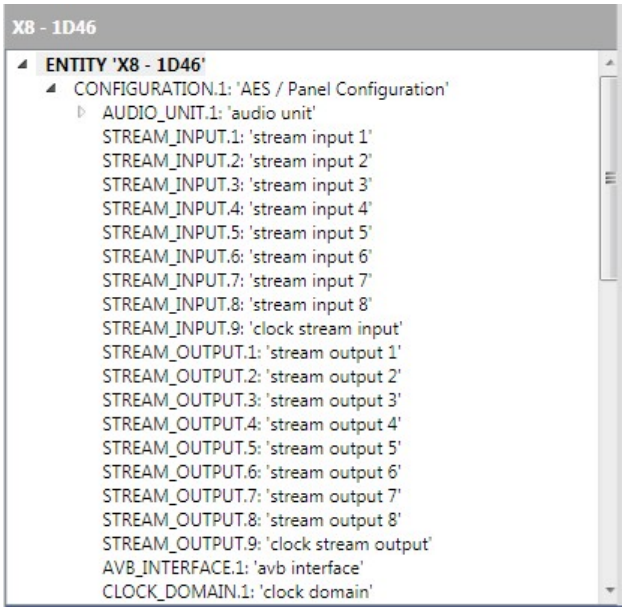


Figure 26: Descriptor groups

The descriptors are grouped as follows:

Entity descriptor	This descriptor describes the highest level of the AVDECC Entity. It repeats some of the information contained within the ADP advertise for the AVDECC Entity as well as the information required to read the rest of the descriptors from the AVDECC Entity.
└ **CONFIGURATION.<n>**	This descriptor describes an AVDECC entity model for a particular setup of the AVDECC entity. The descriptor tells the AVDECC Controller how many of each of the top level descriptors are present in the Configuration.
├ **AUDIO_UNIT.<n>**	This descriptor describes an Audio Unit within the AVDECC entity. An Audio Unit represents a single audio clock domain.
│ ├ **STREAM_PORT_INPUT.<n>** **STREAM_PORT_OUTPUT.<n>**	This descriptor describes a Stream Input or Output Port of the Unit.
│ │ ├ **AUDIO_CLUSTER.<n>**	This descriptor describes groups of audio channels in a Stream. An Audio Cluster could represent a stereo IEC 60958 encoded signal, a one or more channel multi bit linear audio signal, a MIDI signal, or a SMPTE signal.
│ │ └ **AUDIO_MAP.<n>**	This descriptor describes a static mapping between an audio Stream's channels and an Audio Cluster's channels for Streams and Stream Ports that are located in the same Clock Domain.
│ ├ **INTERNAL_PORT_INPUT.<n>** **INTERNAL_PORT_OUTPUT.<n>**	These descriptors describes the end of an internal connection between Units of the AVDECC entity.
│ └ **EXTERNAL_PORT_INPUT.<n>** **EXTERNAL_PORT_OUTPUT.<n>**	These descriptors describes an External Input Port or External Output Port of the unit.
├ **STREAM_INPUT.<n>** **STREAM_OUTPUT.<n>**	These descriptors describes an IEEE Std 1722-2011 sourced or sinked Stream.
├ **AVB_INTERFACE.<n>**	This descriptor describes an interface implementing AVB functionality.
├ **CLOCK_DOMAIN.<n>**	This descriptor describes a source of a common clock signal within an AVDECC entity.
├ **CLOCK_SOURCE.<n>**	This descriptor describes a Clock Source.
├ **JACK_INPUT.<n>** **JACK_OUTPUT.<n>**	These descriptors describes an Input or Output Jack.
├ **MEMORY_OBJECT.<n>**	This descriptor describes a Memory Object representing a region of addressable memory that may be used for settings, log files, or firmware upgrades.
├ **CONTROL_BLOCK.<n>**	This descriptor describes a grouping of Controls within the Configuration or Unit. The Control Block contains an internal signal path and can be used to group a functional set of Controls together.
├ **CONTROL.<n>**	This descriptor describes a generic Control.
└ **LOCALE.<n>**	This descriptor describes a localization of the immutable strings within the AVDECC entity.
└ **STRINGS.<n>**	This descriptor provides up to seven localized strings.

<n> indicates the number of each descriptor and is starting with 1.

One descriptor type can be selected by left-click in the respective row.
The selected row will be highlighted and the available descriptor values will be displayed.
It is not possible to select multiple rows at the same time.

Behind the descriptor is the *Descriptor name* displayed.
The *Descriptor name* is entered in the descriptor values within the group *Object name*.
The *Descriptor name* is also displayed behind descriptor groups, for instance 'testing'.

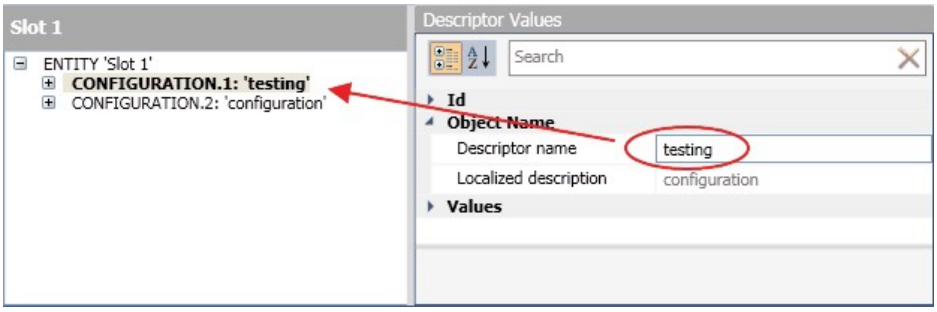


Figure 27: Descriptor name present

If the *Descriptor name* is empty, the *Localized description* is displayed instead.

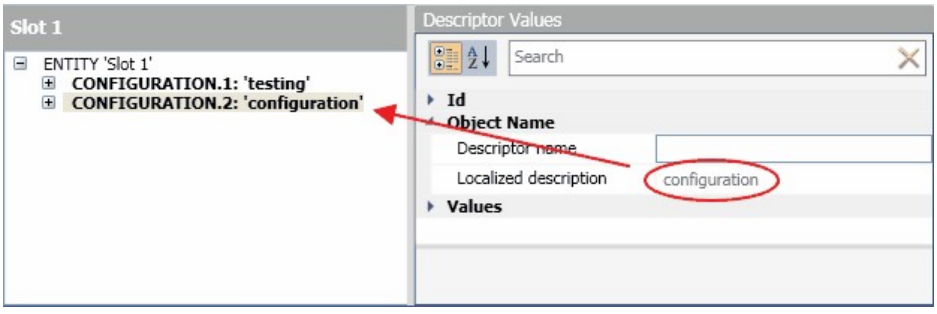


Figure 28: Descriptor name empty

3.2.1.3 Descriptor Values

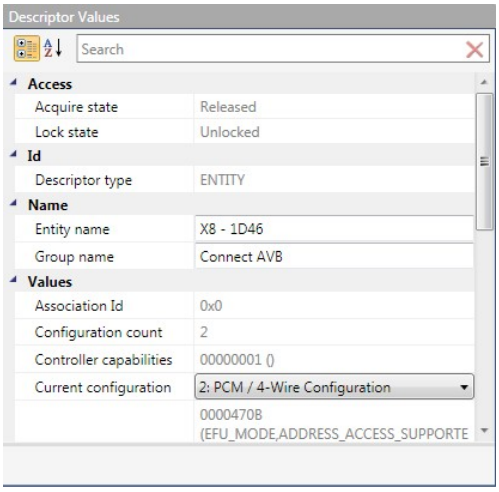


Figure 29: Descriptor-Values

The values can be displayed in two ways:

Icon	Function
	category sorted
	alphabetically sorted

The search field can be used to exclude unwanted values.
Only those kind of values will be displayed, whose names are matching the search criteria.

The descriptor values are separated various groups. Not every group is available in every descriptor.
Following group and values are available in every descriptor group:

Id	
Descriptor index	The Index of the descriptor in the descriptor model.
Descriptor type	The Type of the descriptor in the descriptor model.

Descriptor group specific values are listed below.

ENTITY Descriptor Values

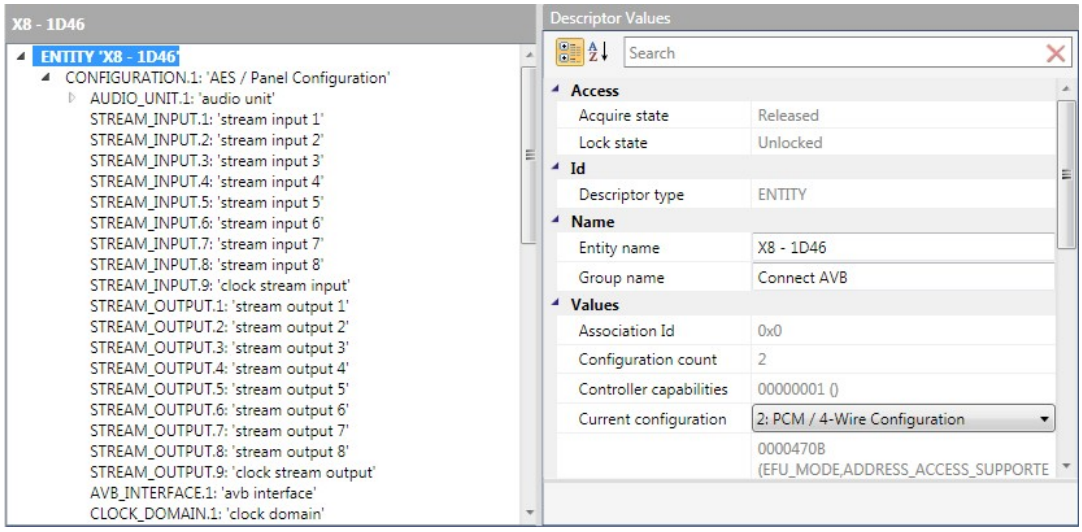


Figure 30: ENTITY Descriptor Values

Access	
Acquire state	The ACQUIRE_ENTITY command is used by a Controller to obtain exclusive access to an entire entity or a subtree of objects. It also functions as a long term lock on the entire entity or a sub-tree of objects.
Lock state	The LOCK_ENTITY command is used to provide short term exclusive access to the entity to perform atomic operations. When an entity is locked it only accepts commands which alter state from the Controller which locked the entity.

Name	
Entity name	Identify the entity be user name. Changes will be written directly to the network.
Group name	Identify multiple entities in a user defined group. Changes will be written directly to the network.

Values	
Association id *	The association_id is used to associate multiple AVDECC entities into a logical collection.
Configuration count	Defines the count of configurations of the entity.
Controller capabilities *	Defines the controller capabilities of the AVDECC entity.
Current configuration	Show the current configuration of the entity. Select a new configuration in the drop down menu.
Entity capabilities *	Defines the The capabilities of the AVDECC entity
Firmware version	64-octet UTF-8 string containing the firmware version of the AVDECC Entity.
Globally unique identifier	Defines the global unique identifier of this entity.
Listener capabilities *	Defines the listener capabilities of the AVDECC entity.
Listener streams	The listener_stream_sinks field is used to identify the number of streams a Listener is capable of simultaneously sinking.
Model id	Identify the entity data model from a vendor.
Model name	The localized string reference pointing to the localized model name.
Serial number	Defines the serial number of the entity.

Talker capabilities *	Defines the talker capabilities of the AVDECC entity.
Talker streams	The talker_stream_sources field is used to identify the number of streams a Talker is capable of sourcing simultaneously.
Vendor id	Identify the manufacturer of the entity.
Vendor name	The localized string reference pointing to the localized vendor name.

* only displayed if expert mode is enabled

CONFIGURATION Descriptor Values

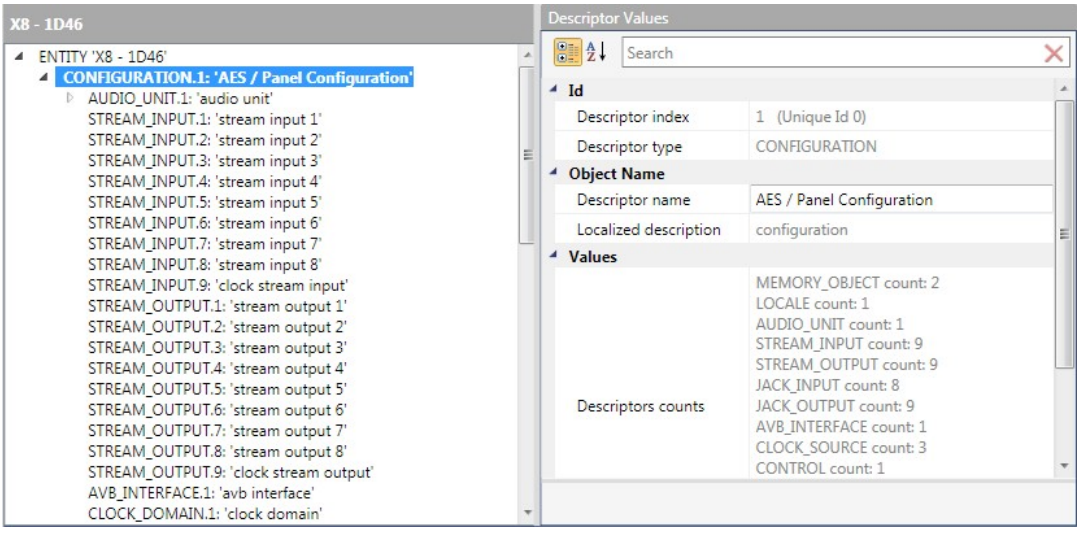


Figure 31: CONFIGURATION Descriptor Values

Descriptor values of the *CONFIGURATION* and of all Descriptors in this group:

Object Name	
Descriptor name	The object name of the descriptor.
Localized description	The index of the descriptor in the descriptor model.

Values *	
Descriptors counts	The number of descriptor counts in the descriptor_counts field. This is referred to as N.
Descriptors counts count	The number of descriptor counts in the descriptor_counts field. This is referred to as N.
Descriptors counts offset	The offset to the descriptor_counts field from the start of the descriptor.

* only displayed if expert mode is enabled

AUDIO_UNIT Descriptor Values

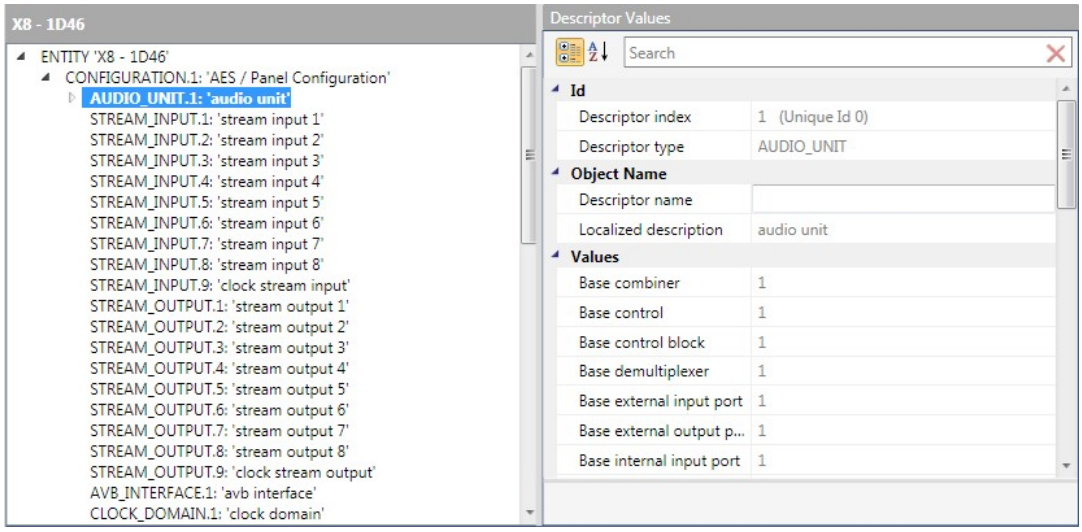


Figure 32: AUDIO_UNIT Descriptor Values

Values	
Base combiner *	The index of the first SIGNAL_COMBINER descriptor.
Base control *	The index of the first CONTROL descriptor.
Base control block *	The index of the first CONTROL_BLOCK descriptor.
Base demultiplexer *	The index of the first SIGNAL_DEMULTIPLEXER descriptor.
Base external input port	The index of the first input EXTERNAL_PORT_INPUT descriptors.
Base external output port	The index of the first output EXTERNAL_PORT_OUTPUT descriptors.
Base internal input port	The index of the first input INTERNAL_JACK_INPUT and INTERNAL_PORT_INPUT descriptors.
Base internal output port	The index of the first output INTERNAL_JACK_OUTPUT and INTERNAL_PORT_OUTPUT descriptors.
Base matrix *	The index of the first MATRIX descriptor.
Base mixer *	The index of the first MIXER descriptor.
Base multiplexer *	The index of the first SIGNAL_MULTIPLEXER descriptor.
Base signal selector *	The index of the first SIGNAL_SELECTOR descriptor.
Base splitter *	The index of the first SIGNAL_SPLITTER descriptor.
Base stream input port	The index of the first STREAM_PORT_INPUT descriptor.
Base stream output port	The index of the first STREAM_PORT_OUTPUT descriptors.
Base transcoder *	The index of the first SIGNAL_TRANSCODER descriptor.
Clock domain index	The descriptor_index of the CLOCK_DOMAIN descriptor describing the clock domain for the unit.
Current sampling rate	The current sample rate of this audio unit.
Number of combiners *	The number of signal combiners within this audio unit.
Number of control blocks *	The number of controls within this audio unit.
Number of controls *	The number of controls within this audio unit.
Number of demultiplexers *	The number of signal demultiplexers within this audio unit.
Number of external input ports	The number of external input ports used by this audio unit.
Number of external output ports	The number of external output ports used by this audio unit.
Number of internal input ports	The number of internal input ports used by this audio unit.
Number of internal output ports	The number of internal output ports used by this audio unit.
Number of matrices *	The number of matrices within this audio unit.
Number of mixers *	The number of mixers within this audio unit.

Number of multiplexers *	The number of signal multiplexers within this audio unit.
Number of signal selectors *	The number of signal selectors within this audio unit
Number of splitters *	The number of signal splitters within this audio unit.
Number of stream input ports	The number of input stream ports used by this audio unit.
Number of stream output ports	The number of output stream ports used by this audio unit.
Number of transcoders *	The number of signal transcoders within this audio unit.
Sampling rates count *	The number of sample rates in the sampling_rates field. The maximum value is 91 - (2*C) for this version of AEM.
Sampling rates offset *	The offset to the sample_rates field from the start of the descriptor.
Supported sampling rates	An array of 4-octet sample rates supported by this audio unit.

* only displayed if expert mode is enabled

STREAM_PORT_INPUT/OUTPUT Descriptor Values

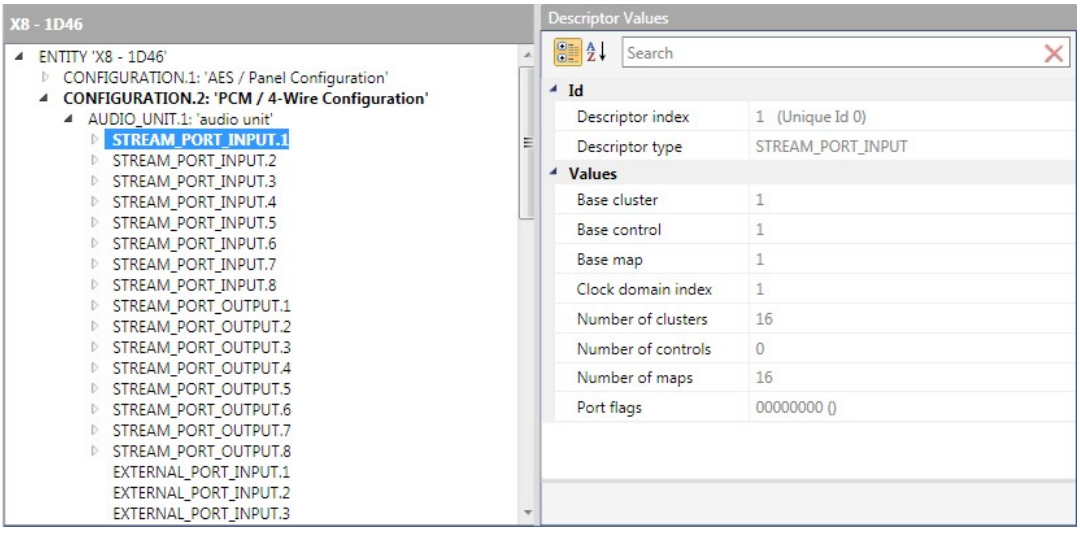


Figure 33: STREAM_PORT_INPUT Descriptor Values

The Values of the *STREAM_PORT_INPUT* Descriptor and *STREAM_PORT_OUTPUT* Descriptor are identical.

Values *	
Base cluster	The index of the first AUDIO_CLUSTER, VIDEO_CLUSTER or SENSOR_CLUSTER descriptor describing the clusters within the port.
Base control	The index of the first CONTROL descriptor.
Base map	The index of the first AUDIO_MAP, VIDEO_MAP or SENSOR_MAP descriptor which defines the mapping between the stream and the port.
Clock domain index	The descriptor_index of the CLOCK_DOMAIN descriptor describing the Clock Domain for the port.
Number of clusters	The number of clusters within the port. This corresponds to the number of AUDIO_CLUSTER, VIDEO_CLUSTER, or SENSOR_CLUSTER descriptors which represent these clusters.
Number of controls	The number of Controls within this Stream Port.
Number of maps	The number of map descriptors used to define the mapping between the stream and the port.
Port flags	Flags describing capabilities or features of the port.

* only displayed if expert mode is enabled

AUDIO_CLUSTER Descriptor Values

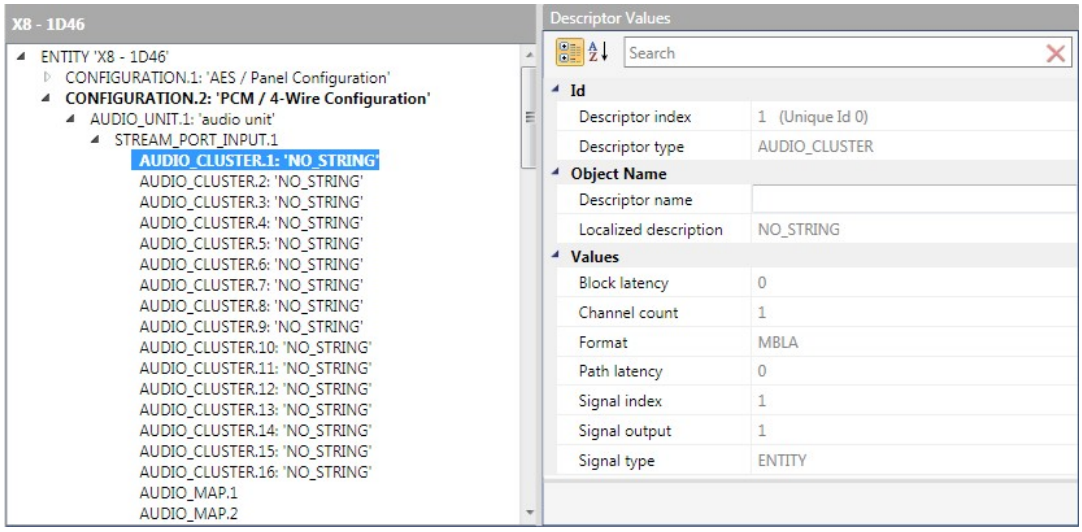


Figure 34: AUDIO_CLUSTER Descriptor Values

Values	
Block latency	For an AUDIO_CLUSTER attached to a STREAM_PORT_INPUT, this is the latency in nanoseconds between the IEEE Std 1722-2011 reference plane and the output of the cluster. For an AUDIO_CLUSTER attached to a STREAM_PORT_OUTPUT, this is the latency in nanoseconds between the output of the previous block's output and the IEEE Std 1722-2011 reference plane. The previous block is the object identified by the signal_type and signal_index fields.
Channel count	The number of channels within the cluster.
Format	The format for each channel of this cluster, all channels within the Cluster have the same format.
Path latency	The latency in nanoseconds between the IEEE 1722-2011 timing reference plane and the opposite end of the currently selected signal path. This does not include any latency added by a DELAY Control. The path_latency field is used to inform smart Controllers of the extra latency to get the samples to the output, so that outputs across multiple entities can be sample aligned.
Signal index	The descriptor_index for the signal source of the cluster.
Signal output	The index of the output of the signal source of the control. For a signal_type of SIGNAL_SPLITTER or SIGNAL_DEMULTIPLEXER this is which output of the object it is being source from, for a signal_type of MATRIX this is the column the signal is from and for any other signal_type this is zero (0).
Signal type	The descriptor_type for the signal source of the cluster.

AUDIO_MAP Descriptor Values

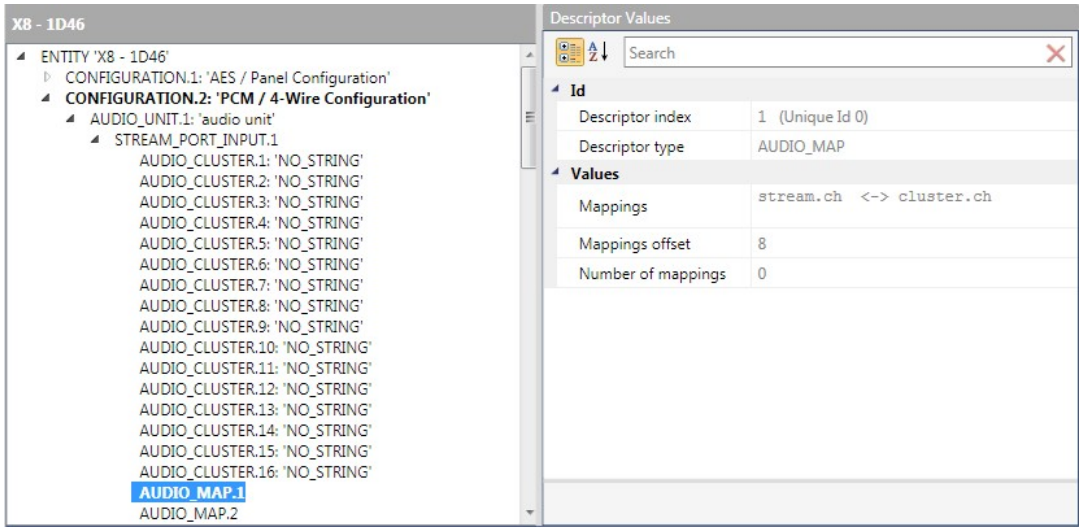


Figure 35: AUDIO_MAP Descriptor Values

Values	
Mappings	The audio channel to stream index and steam channel mappings.
Mappings offset	The offset from the start of the descriptor for the first octet of the mapping_stream_channel[0] string. This field is 8 for this version of AEM.
Number of mappings	The number of channel mappings within the descriptor. The value of this field is referred to as N. The maximum value of this field is 62 for this version of AEM.

INTERNAL_PORT_INPUT/OUTPUT Descriptor Values

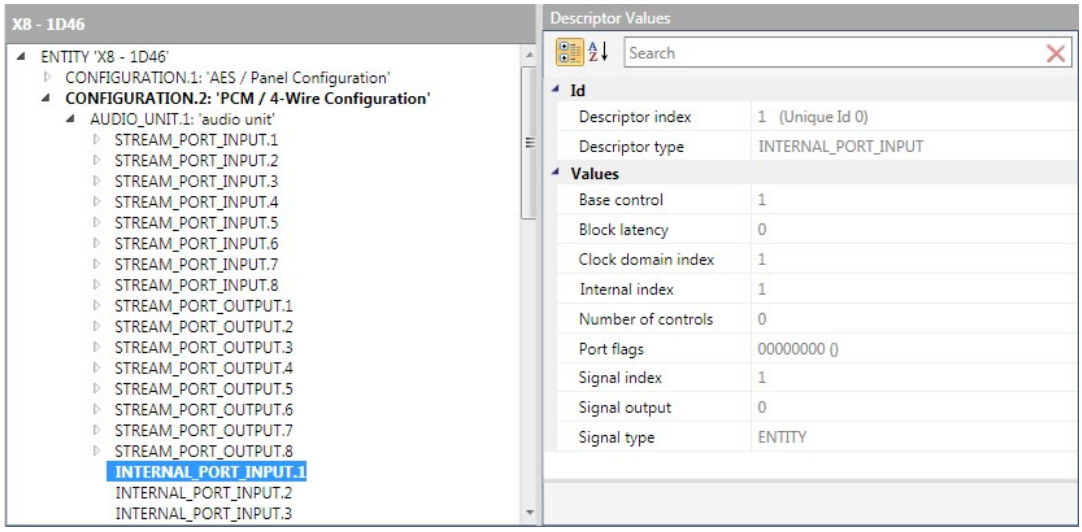


Figure 36: INTERNAL_PORT_INPUT Descriptor Values

The Values of the *INTERNAL_PORT_INPUT* Descriptor and *INTERNAL_PORT_OUTPUT* Descriptor are identical.

Values *	
Base control	The index of the first CONTROL descriptor.
Block latency	For an INTERNAL_PORT_INPUT this is the latency in nanoseconds between the physical pins of the Jack or pads on the PCB and the Port's output. For an INTERNAL_PORT_OUTPUT, this is the latency in nanoseconds between the output of the previous block and the physical pins or pads.
Clock domain index	The descriptor_index of the CLOCK_DOMAIN descriptor describing the Clock Domain for the port.
Internal index	The descriptor_index for the INTERNAL_PORT_INPUT or INTERNAL_PORT_OUTPUT descriptor sourcing or sinking this port on the other unit.
Number of controls	The number of Controls within this External port.
Port flags	Flags describing capabilities or features of the port.
Signal index	The descriptor_index for the signal source of the port.
Signal output	The index of the output of the signal source of the Port. For a signal_type of SIGNAL_SPLITTER or SIGNAL_DEMULTIPLEXER this is which output of the object it is being source from, for a signal_type of MATRIX this is the column the signal is from and for any other signal_type this is zero (0).
Signal type	The descriptor_type for the signal source of the port.

* only displayed if expert mode is enabled

EXTERNAL_PORT_INPUT/OUTPUT Descriptor Values

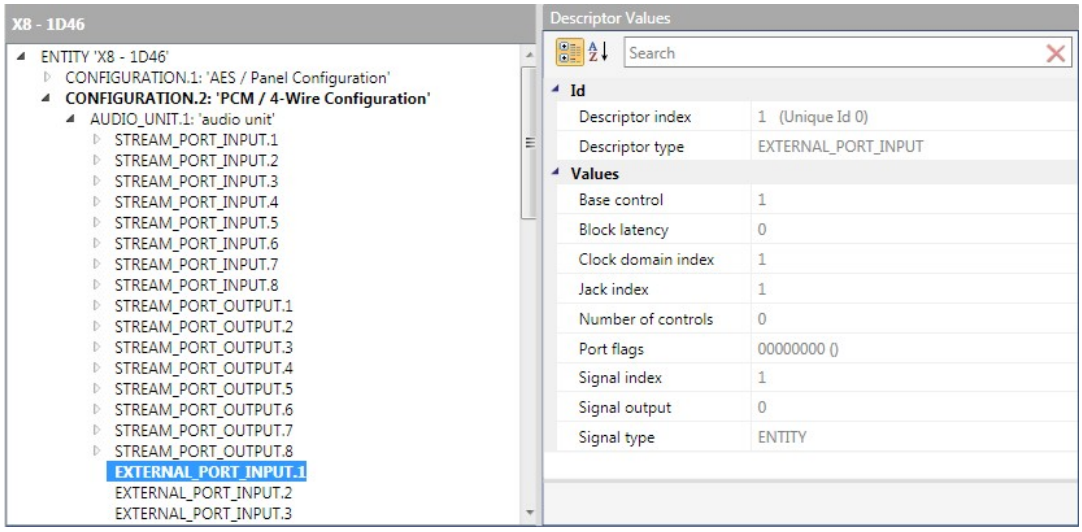


Figure 37: EXTERNAL_PORT_INPUT Descriptor Values

The Values of the *EXTERNAL_PORT_INPUT* Descriptor and *EXTERNAL_PORT_OUTPUT* Descriptor are identical.

Values *	
Base control	The index of the first CONTROL descriptor.
Block latency	For an EXTERNAL_PORT_INPUT this is the latency in nanoseconds between the physical pins of the Jack or pads on the PCB and the Port's output. For an EXTERNAL_PORT_OUTPUT, this is the latency in nanoseconds between the output of the previous block and the physical pins or pads.
Clock domain index	The descriptor_index of the CLOCK_DOMAIN descriptor describing the Clock Domain for the port.
Jack index	The descriptor_index for the JACK_INPUT or JACK_OUTPUT for the port.
Number of controls	The number of Controls within this External port.
Port flags	Flags describing capabilities or features of the port.
Signal index	The descriptor_index for the signal source of the port.
Signal output	The index of the output of the signal source of the Port. For a signal_type of SIGNAL_SPLITTER or SIGNAL_DEMULTIPLEXER this is which output of the object it is being source from, for a signal_type of MATRIX this is the column the signal is from and for any other signal_type this is zero (0).
Signal type	The descriptor_type for the signal source of the port.

* only displayed if expert mode is enabled

STREAM_INPUT/OUTPUT Descriptor Values

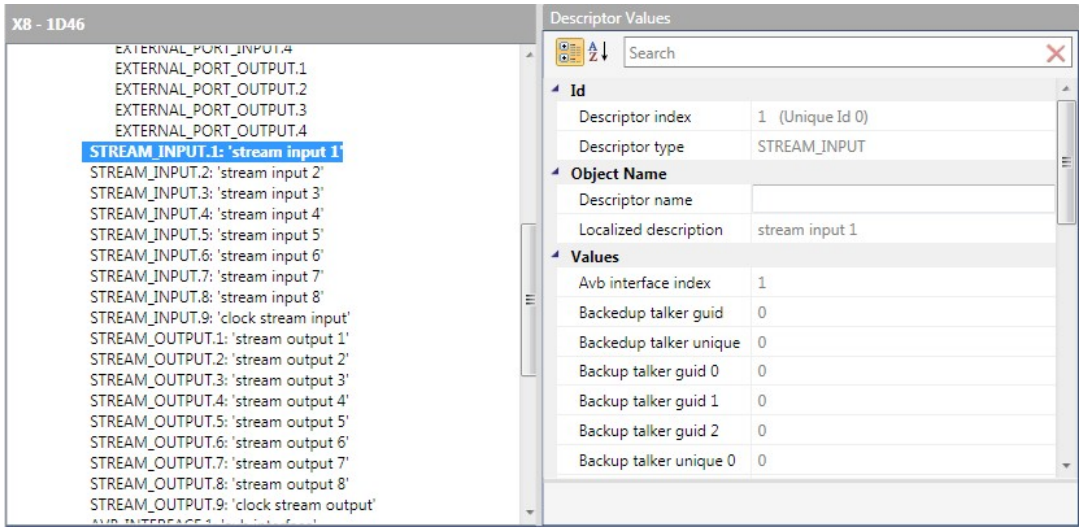


Figure 38: STREAM_INPUT Descriptor Values

The Values of the *STREAM_Input* Descriptor and *STREAM_Output* Descriptor are identical.

Values	
AVB interface index	The descriptor_index of the AVB_INTERFACE from which this stream is sourced or to which it is sinked.
Backed up talker guid *	The GUID of the Talker that this stream is backing up.
Backed up talker unique *	The Unique ID of the Talker that this stream is backing up.
Backup talker guid 0 *	The primary backup Talker's GUID.
Backup talker guid 1 *	The secondary backup Talker's GUID.
Backup talker guid 2 *	The tertiary backup Talker's GUID.
Backup talker unique 0 *	The primary backup Talker's Unique ID.
Backup talker unique 1 *	The secondary backup Talker's Unique ID.
Backup talker unique 2 *	The tertiary backup Talker's Unique ID.
Buffer length *	The length in nanoseconds of the MAC's ingress or egress buffer as defined in IEEE Std 1722-2011 Figure 5.4. For a STREAM_INPUT this is the MAC's ingress buffer size and for a STREAM_OUTPUT this is the MAC's egress buffer size. This is the length of the buffer between the IEEE Std 1722-2011 reference plane and the MAC.
Clock domain index	The descriptor_index of the clock domain providing the media clock for the stream.
Current format	The stream format of the current format.
Formats	Array of stream formats of the supported formats.
Formats offset *	The offset from the start of the descriptor for the first octet of the formats.
Number of formats *	The number of formats supported by this audio stream. The maximum value for this field is 47 for this version of AEM.
Stream flags	Flags describing capabilities or features of the stream.

Values of dynamically stream info *	
Flags	The flags field is set to an appropriate combination of the flags to indicate which fields have values to be set. The lower 16 bits of the flags map directly to the flags field of ACMP.

MSRP accumulated latency	The msrp_accumulated_latency field's use depends on if the command is sent to a STREAM_INPUT or a STREAM_OUTPUT. If it is sent to a STREAM_INPUT, then the msrp_accumulated_latency field is set to the accumulated_latency of the Stream's MSRP Talker Advertise if connected, or zero (0) otherwise. If it is sent to a STREAM_OUTPUT, then the msrp_accumulated_latency field is set to the last set value if it has been set since the Stream was connected, or the appropriate default value for the Stream's traffic class (2 milliseconds for Class A and 50 milliseconds for Class B) if it has not been set since the Stream was connected. The MSRP_ACC_LAT_VALID flag is set only when this field contains a valid accumulated latency.	
MSRP failure bridge_id	The use of the msrp_failure_bridge_id and msrp_failure_code depends on if the command is sent to a STREAM_INPUT or a STREAM_OUTPUT. If it is sent to a STREAM_INPUT, then the msrp_failure_bridge_id and msrp_failure_code fields are set to the failure_bridge_id and failure_code of the Stream's MSRP Talker Failed if the Stream has received an MSRP Talker Failed, otherwise they contain zero (0). If it is sent to a STREAM_OUTPUT, then the msrp_failure_bridge_id and msrp_failure_code fields are set to the last set value if it has been set since the Stream was connected or zero (0) otherwise. The MSRP_FAILURE_VALID flag is set only when these fields contain failure information.	
MSRP failure code		
Stream destination MAC	The stream_dest_mac field is set the destination MAC address of the Stream which has either been previously set or dynamically allocated or zero (00-00-00-00-00-00) if there is no address. The STREAM_DEST_MAC_VALID flag is set only when this field contains a valid destination address.	
Stream format	The stream_format field is set to the current format of the Stream. This is equivalent to the current_format field of the addressed descriptor.	
	Presets	A set of stream format presets. Select an stream format preset to set it to the entity.
	Stream format description	The stream_format field is set to the current format of the Stream. This is equivalent to the current_format field of the addressed descriptor.
Stream id	The stream_id field is set to the current id of the Stream, or zero (0) if the Stream is not connected and has not had a stream_id set. The flags field indicates if this is valid.	
Stream VLAN id	The stream_vlan_id field is set to the VLAN ID of the Stream or zero (0) if the Stream is not connected. The STREAM_VLAN_ID_VALID flag is set only when this field contains a valid VLAN ID.	

* only displayed if expert mode is enabled

AVB_INTERFACE Descriptor Values

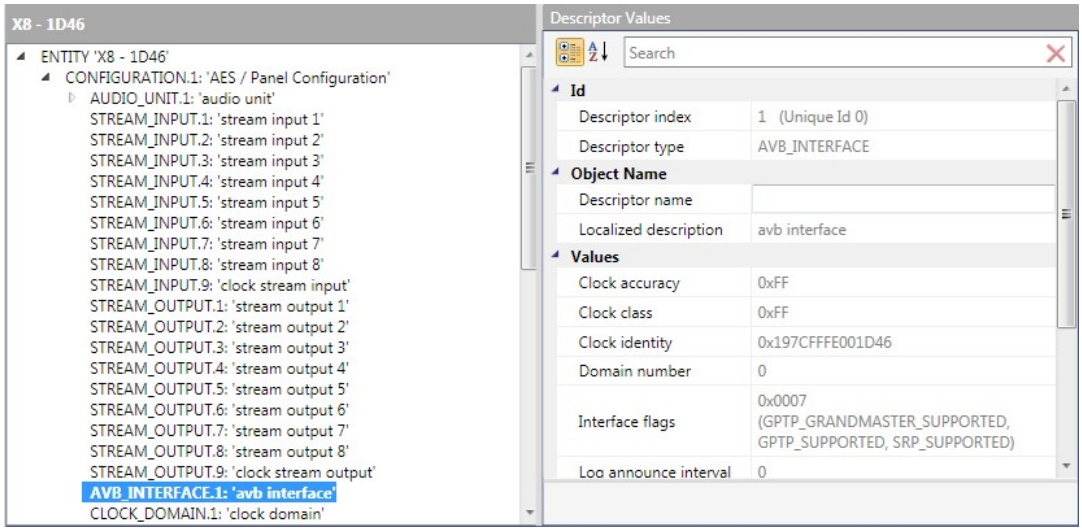


Figure 39: AVB_INTERFACE Descriptor Values

Values	
Clock accuracy	The Clock Accuracy field of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, ff16 otherwise.
Clock class	The Clock Class field of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, ff16 otherwise.
Clock identity	The IEEE Std 802. AS-2011 clock identity of the interface.
Domain number	The Domain Number field of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, zero (0) otherwise.
Interface flags	The flags describing the features of the interface.
Log announce interval	The currentLogAnnounceInterval of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, zero (0) otherwise.
Log pdelay interval	The currentLogPDelayReqInterval of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, zero (0) otherwise.
Log sync interval	The currentLogSyncInterval of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, zero (0) otherwise.
Mac address	The MAC address of the interface.
Offset Scaled log variance	The Offset Scaled Log Variance field of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, 000016 otherwise.
Port number	The Port Number field of the interface as used by IEEE Std 802.1AS-2011 functionality of the AVB interface if supported, 0000 16 otherwise.
Priority1	The Priority1 field of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, ff16 otherwise.
Priority2	The Priority2 field of the IEEE Std 802. AS-2011 grandmaster functionality of the AVB interface if supported, ff16 otherwise.

Values of dynamically AS Path *	
Number of ClockIdentities	The count field is set to the number of ClockIdentities present in the path_sequence field.
Path sequence	The path_sequence field is set to pathSequence of the latest IEEE Std 802.1AS-2011 Announce message PathTrace TLV.

Values of dynamically AVB info *	
Flags	The flags field is set to a combination of values as appropriate from IEEE P1722.1™/D23 March 2013 Table 7.131 or zero (0).
Gptp domain number	The gptp_domain_number field is set to the domainNumber of the current IEEE Std 802.1AS-2011 grandmaster as elected on this AVB Interface.
Gptp grandmaster id	The gptp_grandmaster_id field is set to the ClockIdentity of the current IEEE Std 802.1AS-2011 grandmaster as elected on this AVB Interface.
MSRP mappings	The msrp_mappings field of the GET_AVB_INFO response contains one or more mappings from traffic class to priority and VLAN ID. Offsets are based on the start of the msrp_mappings field. IEEE P1722.1™/D23 March 2013 Table 7.132 shows the msrp_mappings Format.
msrp mapping count	The msrp_mappings_count field is set to the number of mappings present in the msrp_mappings field.
Propagation delay	The propagation_delay field is set to the propagation delay in nanoseconds as reported by the IEEE Std 802.1AS-2011 pDelay mechanism.

* only displayed if expert mode is enabled

CLOCK_DOMAIN Descriptor Values

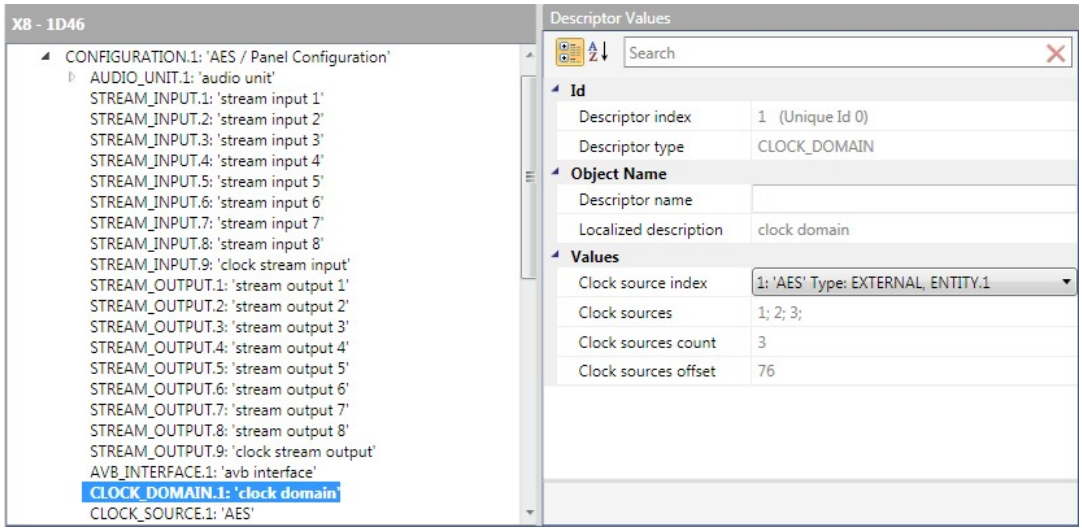


Figure 40: CLOCK_DOMAIN Descriptor Values

Values	
Clock source index	The descriptor_index describes the current clock source for the clock domain. In the current configuration the user may select a new clock source for the clock domain Clock Source Types: INTERNAL: The clock is sourced within the entity such as from a crystal oscillator. EXTERNAL: The clock is sourced from an external connection on the entity (via a Jack). INPUT_STREAM: The clock is sourced from the media clock of an Input Stream. If clock Source Type of INPUT_STREAM, the corresponding Input Stream must have set the 'Clock_Sync_Source' flag.see 'AVnu PA1 Pro Audio Media Clocking Specification' document.
Clock sources *	The list of CLOCK_SOURCE descriptor indices which the clock_source_index may be set to.
Clock sources count *	The number of Clock Source indexes in the clock_sources field. The value of this field is referred to as C. The maximum value for this field is 249 for this version of AEM.
Clock sources offset *	The offset to the clock_sources field from the start of the descriptor. This is 76 for this version of AEM.

* only displayed if expert mode is enabled

CLOCK_SOURCE Descriptor Values

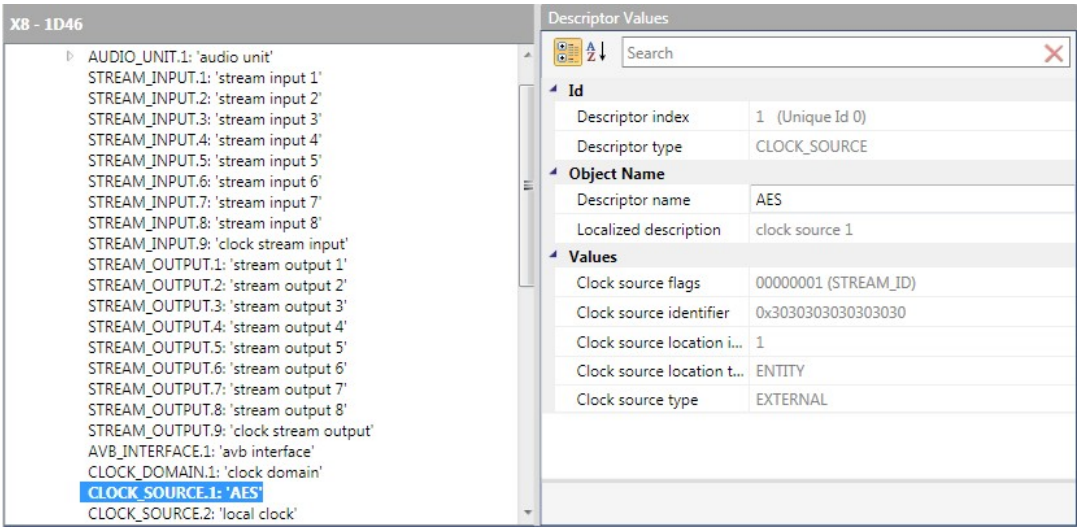


Figure 41: CLOCK_SOURCE Descriptor Values

Values	
Clock source flags *	Flags describing capabilities or features of the clock source.
Clock source identifier *	The GUID of the source for this clock.
Clock source location index *	The descriptor_index of the object that this clock source is associated with.
Clock source location type *	The descriptor_type of the object that this clock source is associated with.
Clock source type	The type of clock source. INTERNAL: The clock is sourced from within th entity such as from a crystal oscillator. EXTERNAL: The clock is sourced from an external connection on the entity (via a Jack). INPUT_STREAM: The clock is sourced from the media clock of an Input Stream.

* only displayed if expert mode is enabled

JACK_INPUT/OUTPUT Descriptor Values

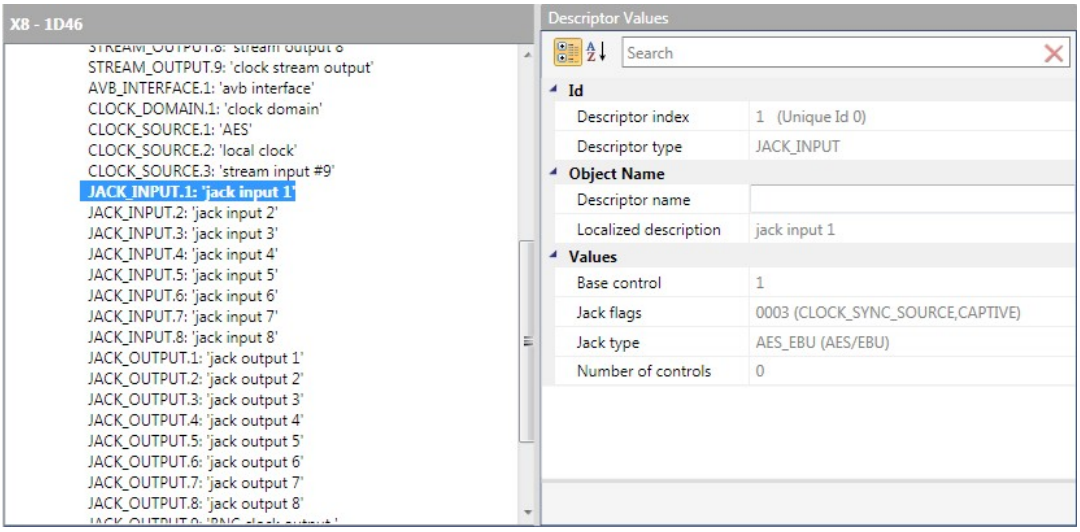


Figure 42: JACK_INPUT Descriptor Values

The Values of the *Jack_Input* Descriptor and *Jack_Output* Descriptor are identical.

Values	
Base control *	The index of the first CONTROL descriptor.
Jack flags *	Flag describing capabilities or features of the Jack.
Jack type	The type of the jack.
Number of controls *	The number of controls within this jack.

* only displayed if expert mode is enabled

MEMORY_OBJECT Descriptor Values

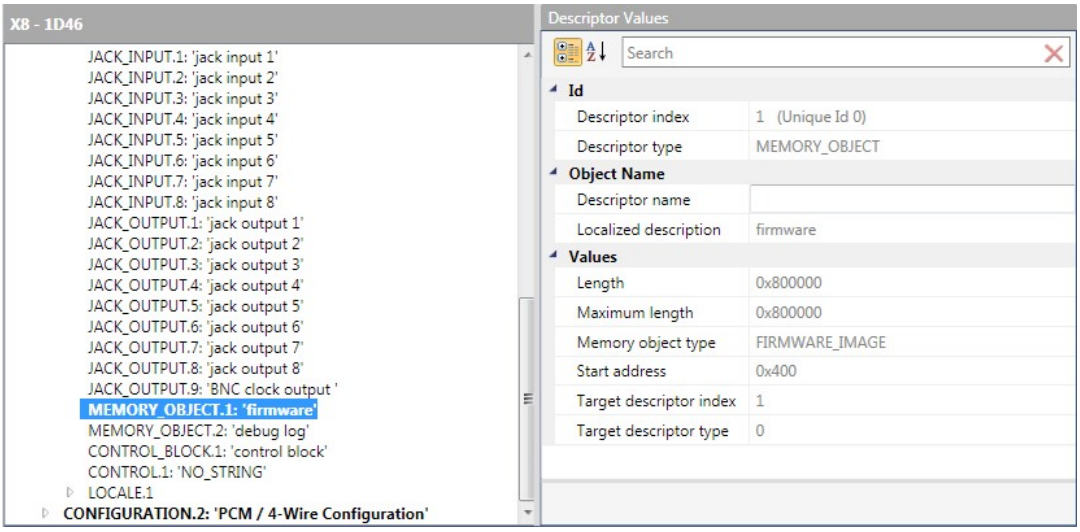


Figure 43: MEMORY_OBJECT Descriptor Values

Values	
Length *	The 64 bit actual length of the memory object. This value will change and will reflect the actual size of the data contained in the memory region described by this memory object.
Maximum length *	The 64 bit maximum length of the memory object.
Memory object type	The type of the memory object.
Start address *	The 64 bit start address used for reading or writing the object's data.
Target descriptor index *	The descriptor_index of the object that is the target of the memory region. This is the object that the settings, log file or firmware applies.
Target descriptor type *	The descriptor_type of the object that is the target of the memory region. This is the object that the settings, log file or firmware applies.

* only displayed if expert mode is enabled

CONTROL_BLOCK Descriptor Values

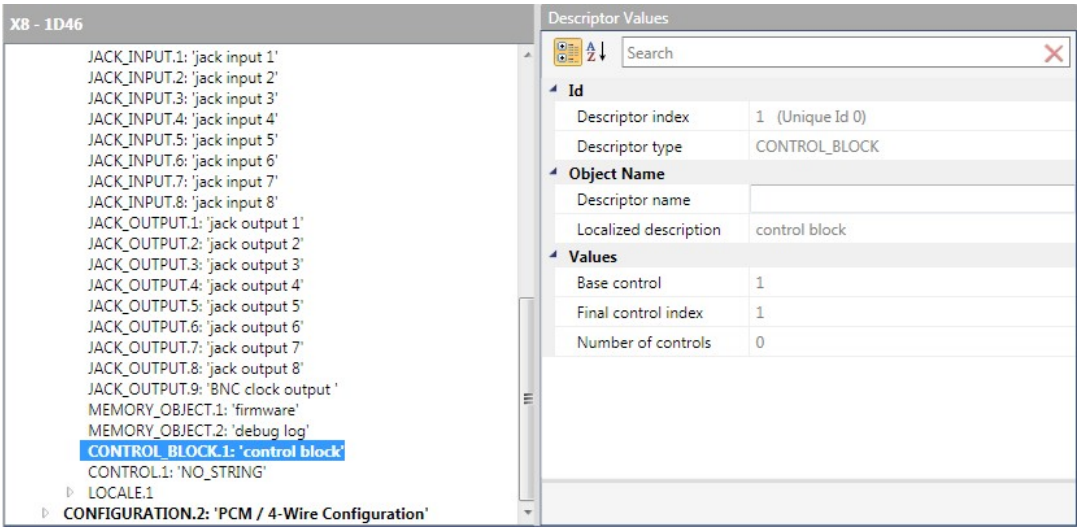


Figure 44: CONTROL_BLOCK Descriptor Values

Values	
Base control	The index of the first CONTROL descriptor.
Final control index	The index of the final CONTROL descriptor in the internal signal chain. If there is no internal signal chain, then this is set to zero (0).
Number of controls	The number of Control within this Control Block.

CONTROL Descriptor Values

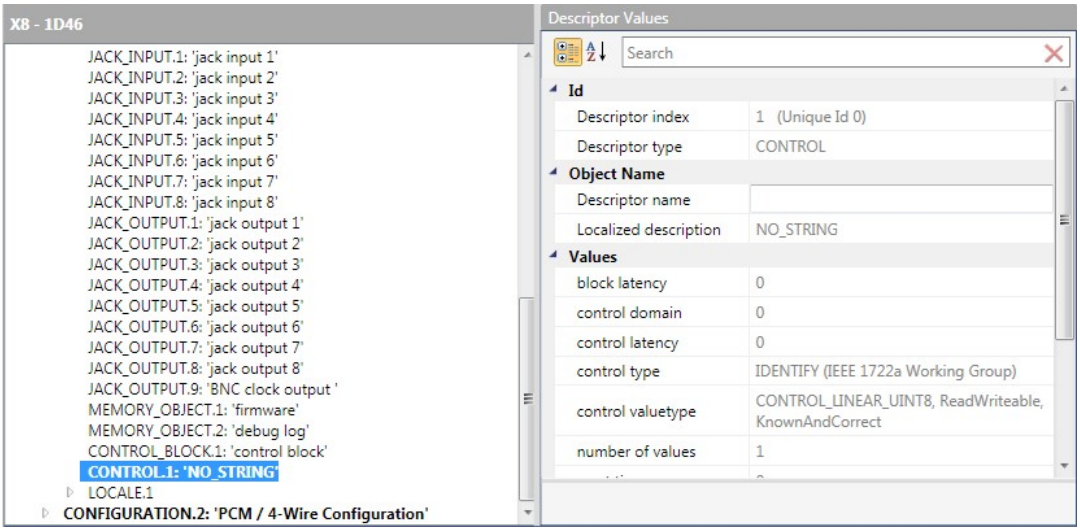


Figure 45: CONTROL Descriptor Values

Values	
block latency	This is the latency in nanoseconds between the output of the previous block and it's output. The previous block is the object identified by the signal_type and signal_index fields. For a DELAY control, the value of the delay is not included in this value.
control domain	The domain that this control belongs to. An AVDECC Entity may have one or more control domains to restrict access to controls. By default an AVDECC Entity uses control domain 0 for all controls.
control latency	The worst case time in microseconds from when a control value change is received and when the control has completely switched to the new value.
control type	The type of the control. See Table 7.94 for the table of valid control types.
control valuetype	The type of the value contained in the control as defined in 7.3.5.1 . The control value type determines T, the size of a value entry in the value_details array.
number of values	The number of value settings this control has. The value of this field is referred to as N. The maximum value of this field is defined in the 'Max Value Count' column of Table 7.39.
reset time	The time period in microseconds from when a control is set with the SET_CONTROL command till it automatically resets to it's default values. When this is set to zero (0) automatic resets do not happen.
signal index	The descriptor_index for the signal source of the control.
signal output	The index of the output of the signal source of the control. For a signal_type of SIGNAL_SPLITTER or SIGNAL_DEMULTIPLEXER this is which output of the object it is being source from, for a signal_type of MATRIX this is the column the signal is from and for any other signal_type this is zero (0).
signal type	The descriptor_type for the signal source of the control.
values offset	The offset from the start of the descriptor for the first octet of the value_details. This field is 104 for this version of AEM.

Values details	
value detail	control specific value details

LOCALE Descriptor Values

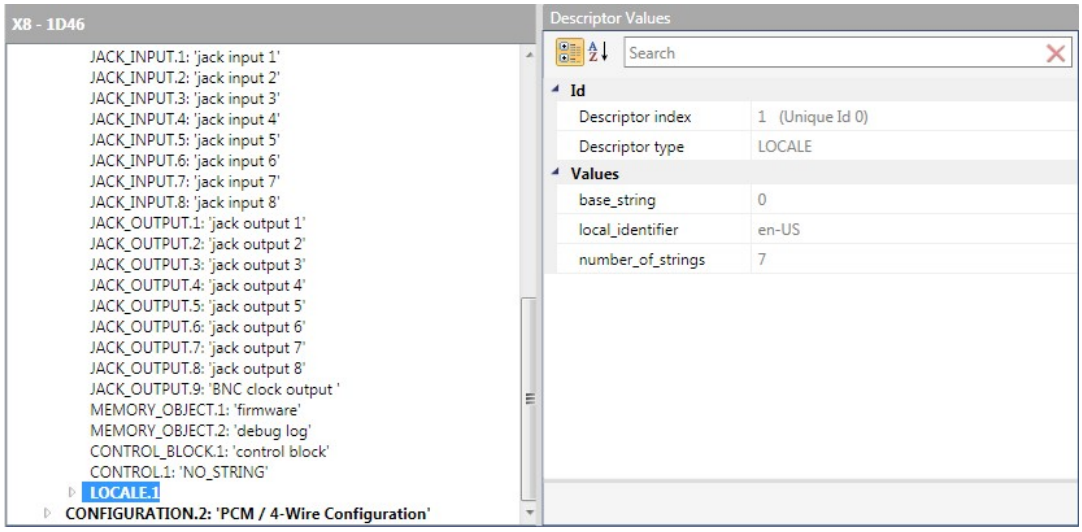


Figure 46: LOCALE Descriptor Values

Values	
base_string *	The descriptor index of the first STRINGS descriptor for this locale.
local_identifier	64 octet UTF-8 string containing the locale identifier.
number_of_strings *	The number of STRINGS descriptors in this locale. This es the same value for all locales in an AVDECC Entity.

* only displayed if expert mode is enabled

STRINGS Descriptor Values

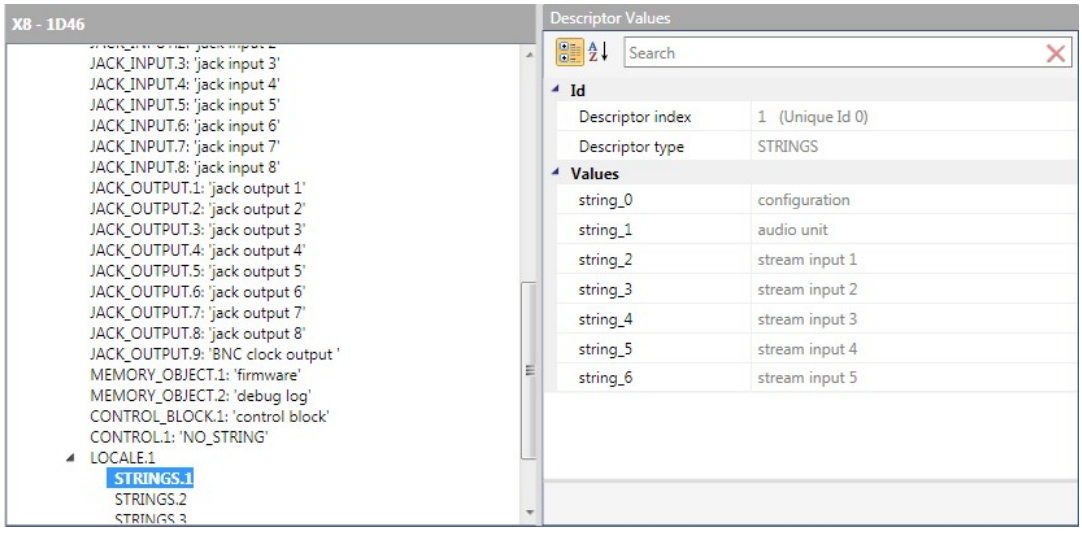


Figure 47: STRINGS Descriptor Values

Values	
string_x	64 octet UTF-8 string at index x.

3.2.2 Connection Management

The Connection Management tab contains 2 areas.

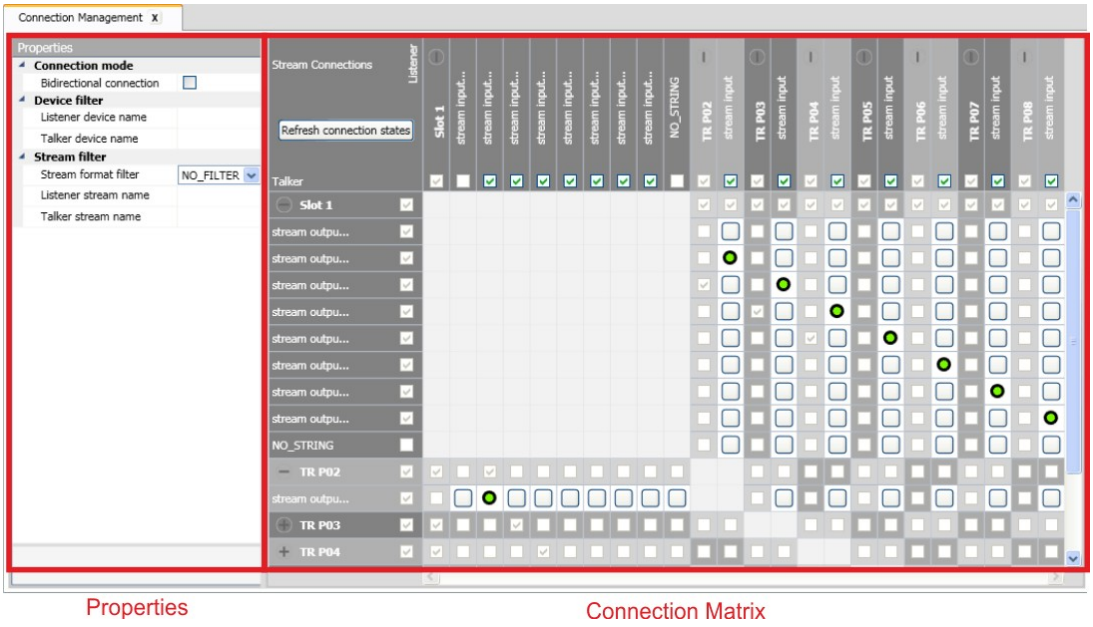


Figure 48: Connection Management (click to jump to the respective chapter)

3.2.2.1 Properties

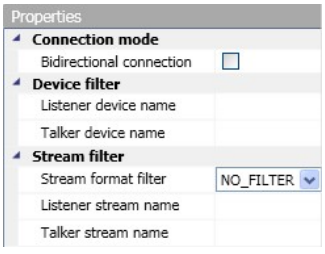


Figure 49: Connection Management Properties

The filter fields can be used to exclude unwanted entities.
Only those entities will be displayed, whose names are matching the search criteria.
Use '*' for wildcard searching. Example: Use 'Dev*' or '*X8' or '*vice*' to find the name 'Device-X8'.
Use '|' for OR-operation. Example: 'name1 | name2'

The Properties contains following fields:

Connection mode	
Bidirectional connection	If enabled, the manager tries to establish a bidirectional connection by default.

Device Filter	
Listener Device Name	Filter by listener device name to only show entities of filter categories.
Talker Device Name	Filter by talker device name to only show entities of filter categories.
Use regular expressions	If enabled, the manager interprets arbitrarily substrings in the filter field.

Stream Filter (only streams with matching format will be displayed)		
Stream format filter	No Filter	No Filter applied. Show all streams.
	AES Audio Format	Show only AES Audio Format streams.
	Mono PCM Audio Format	Show only Mono PCM Audio Format streams.
	Dual PCM Audio Format	Show only Dual PCM Audio Format streams.
	Only Clock Streams	Show only Clock Stream streams.
Listener stream name	Filter by listener stream name to only show entities of filter categories.	
Talker stream name	Filter by talker stream name to only show entities of filter categories.	
Use regular expressions	If enabled, the manager interprets arbitrarily substrings in the filter field.	

3.2.2.2 Connection Matrix

The Connection Matrix contains 3 areas:

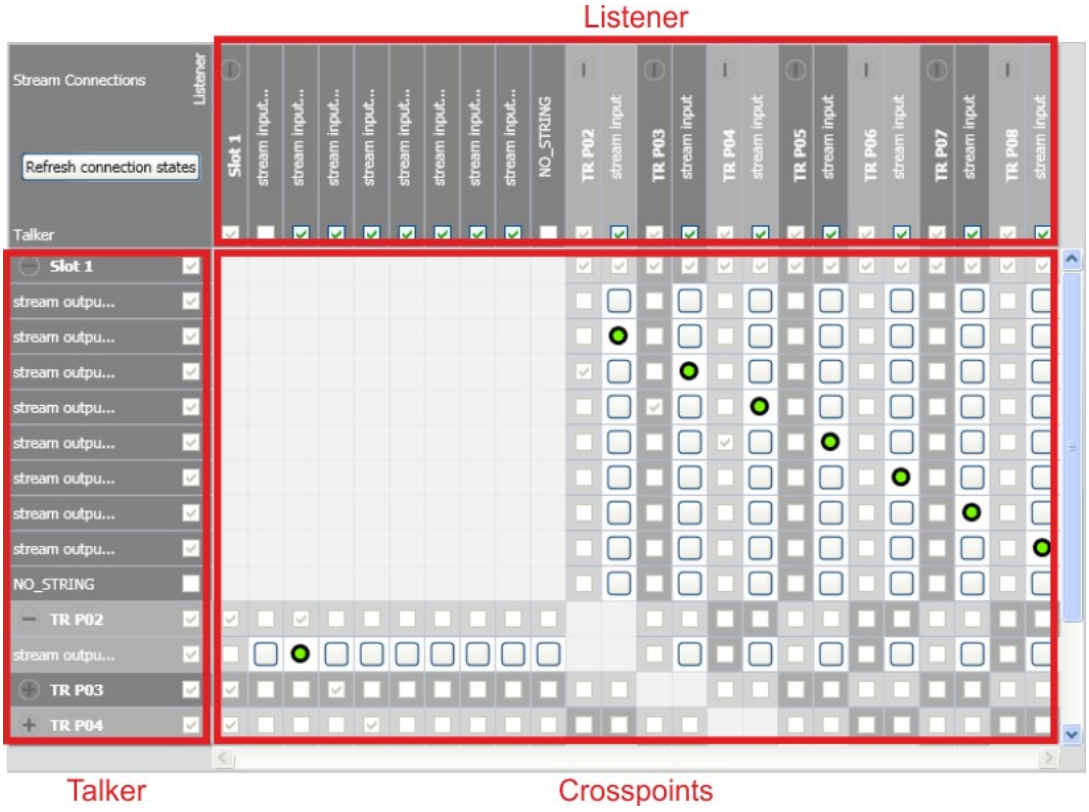


Figure 50: Connection Matrix

All Talkers of the AVB network are displayed vertically on the left side of the Connection Matrix.
All Listeners are displayed horizontally on the top of the Connection Matrix.
Depending of the option setting *Expand Matrix per default*, the single channels of the entities are collapsed or expanded. Expanding and collapsing is done by clicking on following symbols:

Symbol	Meaning	
⊕	Collapsed device view	Only the name of entity is displayed.
−	Expanded device view	All channels of the entity are displayed.

The checkboxes next to the Talkers/Listeners name informs about at least one present connection. This checkbox is *only* for information purpose and can *not* be checked/unchecked.
Exceptions are the checkboxes of the single channels of the Listeners. A checked checkbox can be unchecked by clicking on it. This will disconnect the Listener from the respective Talker.

Symbol	Meaning	
	Unchecked field	No connection set up.
	Checked field	At least one connection is set up.
	Checked field (single channel of Listener)	At least one connection is set up. The Listener can be disconnected by unchecking this checkbox.

The crosspoints between the Talker and the Listeners informs about the connection between the respective entities. Following symbols are used:

Symbol	Meaning	
	Unconnected Crosspoint	The Listener is *not* connected to the Talker.
	Connected Crosspoint	The Listener is connected to the Talker.
	Connection Error	Hovering over this item will show a tooltip about the error.
	Unknown Status	The system wasn't able to read the Status of the entity.

Crosspoints

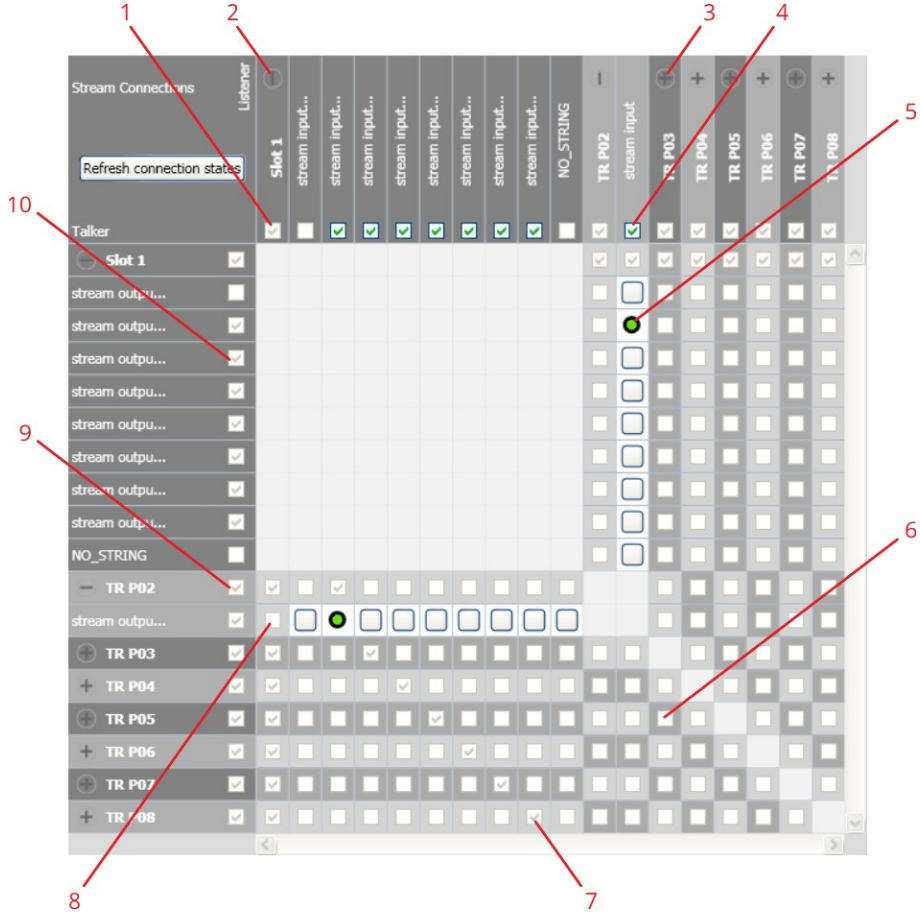


Figure 51: Crosspoints

1	This checkbox indicates the connection status of the Listener entity. The checkbox is checked if the Listener entity has at least one connection to a Talker. In this case Listener entity '*Slot 1*' is connected to a Talker.
2	Expanded view of Listener entity '*Slot 1*'. The single streams of the entity are visible.
3	Collapsed view of Listener entity '*TR P03*'. Only the entity is visible, the single channels are hidden.
4	This checkbox indicates the connection status of the Listener channel. The checkbox is checked if the Listener channel has a connection to a Talker. In this case channel '*stream input*' of Listener '*TR P02*' is connected to a Talker. This connection can be disconnected by unchecking this box. The connection state results of following command response: 'GET_RX_STATE_RESPONSE', 'DISCONNECT_RX_RESPONSE' or 'CONNECT_RX_RESPONSE'.
5	This checkbox indicates the connection status between the Talker stream and the Listener channel. The checkbox is checked if the Talker stream is connected to the Listener channel. In this case stream '*stream output 2*' of Talker '*Slot 1*' is connected to Listener channel '*stream input*'. This connection can be disconnected by unchecking this box.
6	This checkbox indicates the connection status between the Talker and Listener entity. The checkbox is checked if there is at least one connection between the Talker and Listener entity. In this case Listener entity '*TR P03*' is not connected to Talker entity '*TR P05*'. The connection state results of following command response: 'GET_RX_STATE_RESPONSE', 'DISCONNECT_RX_RESPONSE' or 'CONNECT_RX_RESPONSE'.
7	This checkbox indicates the connection status between the Listener channel and the Talker entity. The checkbox is checked if the Listener channel has a connection to the Talker entity. In this case channel '*stream input 8*' of Listener '*Slot 1*' is connected to Talker entity '*TR P08*'. The connection state results of following command response: 'GET_RX_STATE_RESPONSE', 'DISCONNECT_RX_RESPONSE' or 'CONNECT_RX_RESPONSE'.
8	This checkbox indicates the connection status between the Talker stream and the Listener entity. The checkbox is checked if the Talker stream has at least one connection to the Listener entity. In this case stream '*stream output 1*' of Talker '*TR P02*' is connected to Listener entity *'Slot 1'*. The connection state results of following command response: 'GET_TX_STATE_RESPONSE', 'GET_TX_CONNECTION_RESPONSE'.
9	This checkbox indicates the connection status of the Talker entity. The checkbox is checked if the Talker entity has at least one connection to a Listener. In this case Talker entity '*TR P02*' is connected to a Listener.
10	This checkbox indicates the connection status of the Talker stream. The checkbox is checked if the Talker stream has at least one connection to a Listener. In this case stream '*stream output 3*' of Talker '*Slot 1*' is connected to a Listener. The connection state results of following command response: 'GET_TX_STATE_RESPONSE'.

	A Listener channel can be connected to only one Talker stream at the same time. A Talkers stream can be connected to multiple Listener channels, if supported by the entity.

Tooltips

Hovering over an entity shows a tooltip:

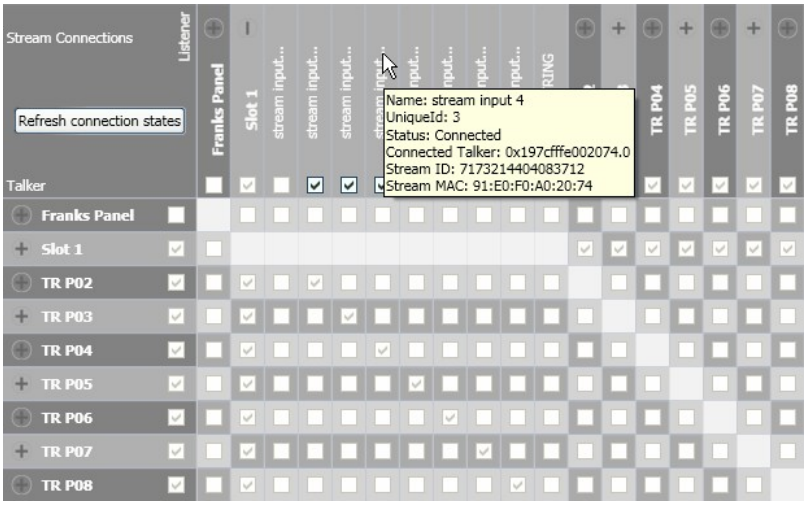


Figure 52: Tooltip

The contents of the tooltip depends on the type of entity (Talker/Listener/Entity/Channels):

Entities (Talkers/Listener)	
Name	Name of the entity.
Groupname	Name of the group of the entity.
GUID	Global Unique Identifier of the entity.
Version	Software Version of the entity.
Grandmaster	Grandmaster Id of the entity.
Vendor	Entity vendor name
Model	Entity model name

Channels (of Talker/Listener)	
Name	Name of the entity.
Unique Id	The Index of the descriptor in the descriptor model.
Status	Connected/Disconnected
Connected Talker (Listener Channels only)	Entity Id of the connected Talker
Stream ID	The stream_id field is set to the current id of the Stream, or zero (0) if the Stream is not connected and has not had a stream_id set. The flags field indicates if this is valid.
Stream MAC	The stream_dest_mac field is set the destination MAC address of the Stream which has either been previously set or dynamically allocated or zero (00-00-00-00-00-00) if there is no address. The STREAM_DEST_MAC_VALID flag is set only when this field contains a valid destination address.
Connected Listener (Talker Channels only)	Number of connected Listeners and Entity Id of every connected Listener.

Change Object Name

A right mouse click on the stream names shows a dialog window to change directly the object name of the respective entry.

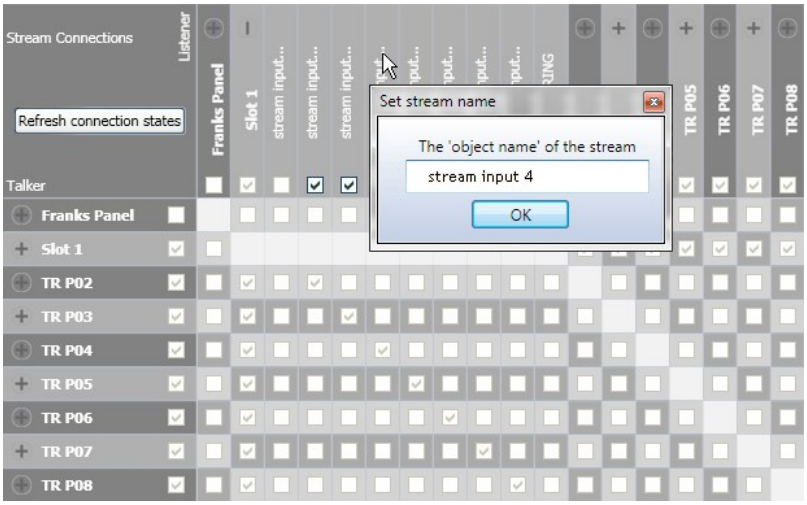


Figure 53: Change Object Name

3.2.3 Firmware Update

The Firmware Update tab contains 2 areas:

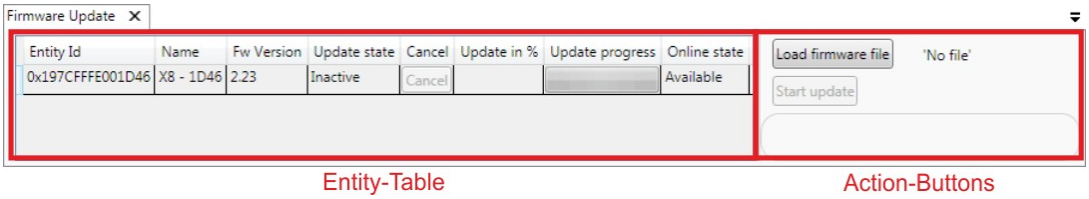


Figure 54: Firmware Update (click to jump to the respective chapter)

3.2.3.1 Entity-Table

The table of entities contains 8 columns:

Column	Content	
Entity Id	Global Unique Identifier of the entities Only entities with following conditions will appear: • Entity must support a firmware update • Entity must have the memory descriptor 'firmware_image'	
Name	Name of the entity	
Fw Version	The actual Firmware Version	
Update State	Inactive	No Update process ongoing
	UploadStarted	'UPLOAD' command has been sent
	UploadInProgress	'UPLOAD' response has been received
	StoreStarted	'STORE' command has been sent
	StoreInProgress	'STORE' response has been received
	StoreFinished	'OPERATION_STATUS" has reached 100%
	Aborting	'ABORT' command has been sent
	Aborted	'ABORT' response has been received
	Error	Update aborted after error
Cancel	Action button to cancel the firmware update	
Update in %	Display of the actual Update progress (percentage)	
Update progress	Display of the actual Update progress (horizontal bar)	
Online State	State of the stream	
	Available	Entity is available in the AVB network
	Departing	Entity departed in the AVB network and is no longer available
	Timeout	Entity doesn't send alive signal any more and is no longer available

Multiple entities can be selected by left-click in the respective rows while pressing *SHIFT* or *CTRL* on the keyboard. The selected row will be highlighted.

A popup menu appears by right-click on an entity:

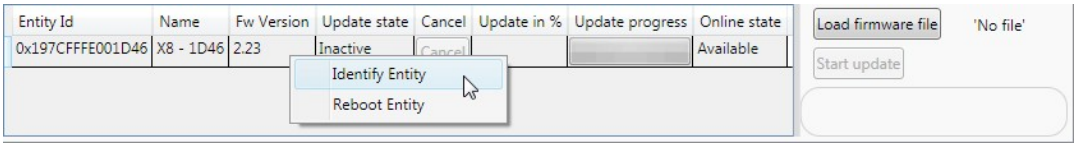


Figure 55: Right-click on Entity-Table

Command	Function
Identify Entity	The respective entity signalize itself to be identified by the operator. The signalisation is device dependent, i.e. all LEDs will flash for a few seconds.
Reboot Entity	The selected entity will be rebooted.

3.2.3.2 Action-Buttons

There are 2 action buttons:

Button	Action
Load Firmware file	Opens a window to browse to the location of the firmware file
Start Update	Starts the Update procedure on all selected entities

4 How to ...

4.1 setup/delete connections

➢ Select 'Connection Management' in the View-Selector.

Figure 56: View-Selector

➢ Click the Collapse/Expand-Symbols of the desired entities of Talker and Listener.
(for instance Talker 'TR P04' and Listener 'Slot 1').

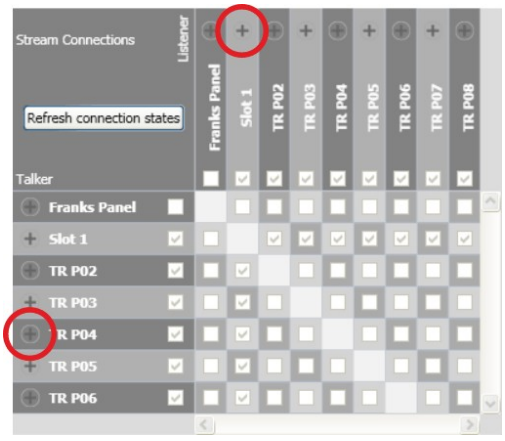


Figure 57: Collapse/Expand-Symbols

➢ Click an empty Crosspoint to establish the connection.
(for instance connection between channel 'output stream 1' and 'stream input 8').
or
➢ Click a present Crosspoint to delete the connection.
(for instance connection between channel 'output stream 1' and 'stream input 4').

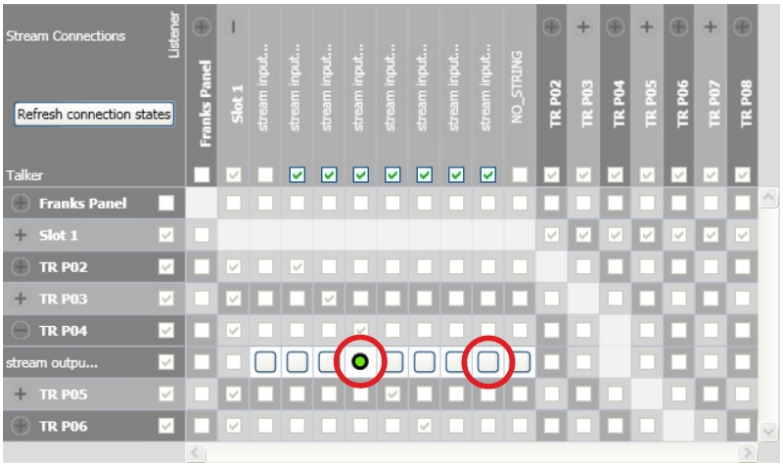


Figure 58: establish/delete connection

4.2 select a configuration

An entity can have multiple sets of configurations.
The current configuration selection can be changed as follows:

➢ Select 'Entity properties' in the View-Selector.

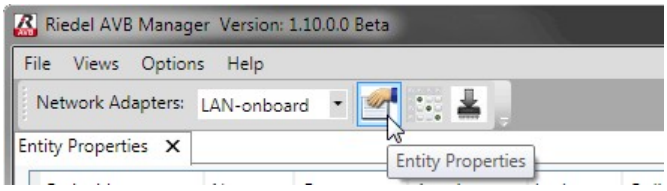


Figure 59: View-Selector

➢ Select the desired entity in the Entity-Table.

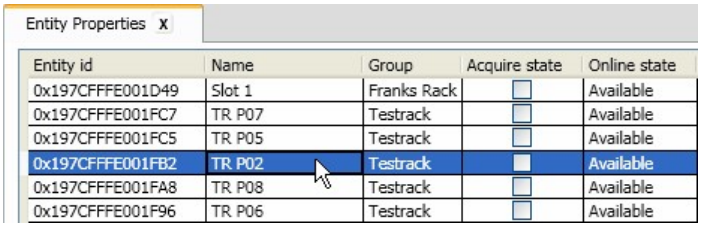


Figure 60: Table of entities

➢ Select the main Descriptor ENTITY.

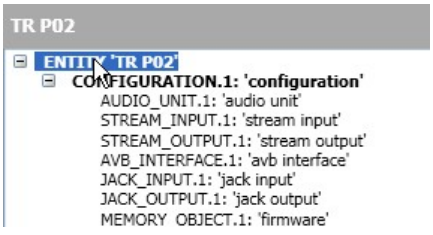


Figure 61: Descriptor area

➢ Select in the Descriptor Values '*Current configuration*' in the group '*Values*' the desired configuration in the drop down menu.

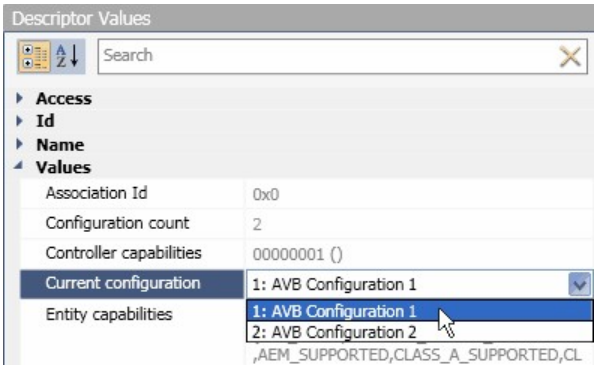


Figure 62: Descriptor Values area

4.3 setup clocking

➢ Select 'Entity properties' in the View-Selector.

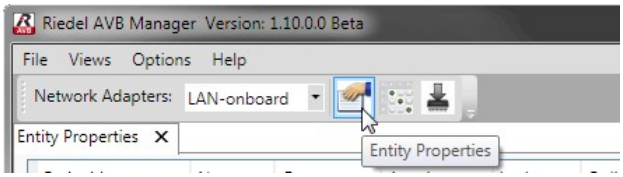


Figure 63: View-Selector

➢ Select the desired entity in the Entity-Table

Entity id	Name	Group	Acquire state	Online state	Identified at	Grandmaster id
0x197CFFFE001D49	Slot 1	Franks Rack	☐	Available		0x496FFFE8389D9
0x197CFFFE001FC7	TR P07	Testrack	☐	Available		0x496FFFE8389D9
0x197CFFFE001FC5	TR P05	Testrack	☐	Available		0x496FFFE8389D9
0x197CFFFE001FB2	TR P02	Testrack	☐	Available		0x496FFFE8389D9
0x197CFFFE001FA8	TR P08	Testrack	☐	Available		0x496FFFE8389D9
0x197CFFFE001F96	TR P06	Testrack	☐	Available		0x496FFFE8389D9

Figure 64: Table of entities

➢ Select the CLOCK_DOMAIN Descriptor.

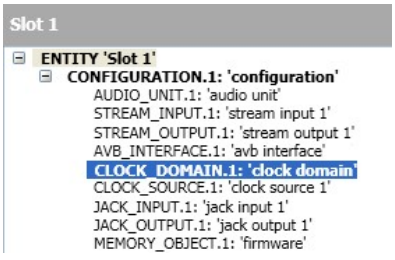


Figure 65: Descriptor area

➢ Select in the Descriptor Values '*Clock source index*' in the group '*Values*' the desired clock input in the drop down menu.

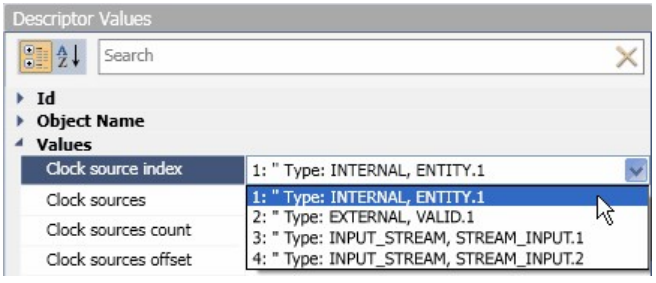


Figure 66: Descriptor Values area

Clock Source Types	
INTERNAL	The clock is sourced within the entity such as from a crystal oscillator.
EXTERNAL	The clock is sourced from an external connection on the entity (via a Jack).
INPUT_STREAM	The clock is sourced from the media clock of an Input Stream.

4.4 update the firmware

➢ Select 'Firmware Update' in the View-Selector.

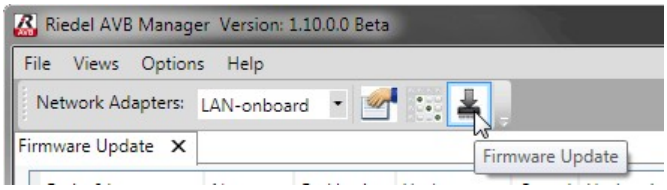


Figure 67: View-Selector

➢ Click the button *Load firmware file*.

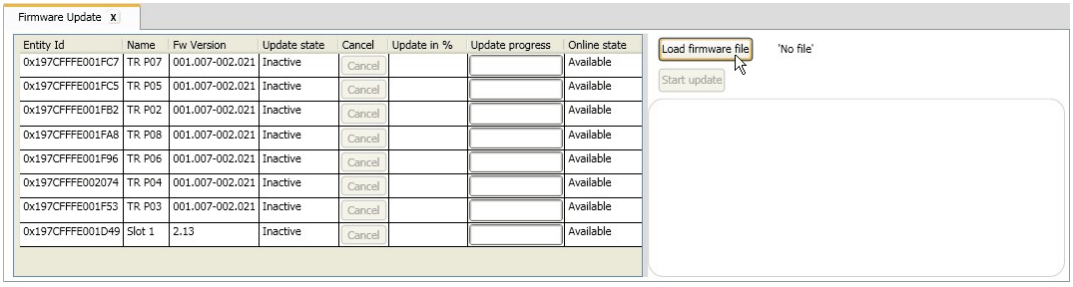


Figure 68: load firmware file

➢ Navigate to the path of the firmware and select the *firmware.upg* file.

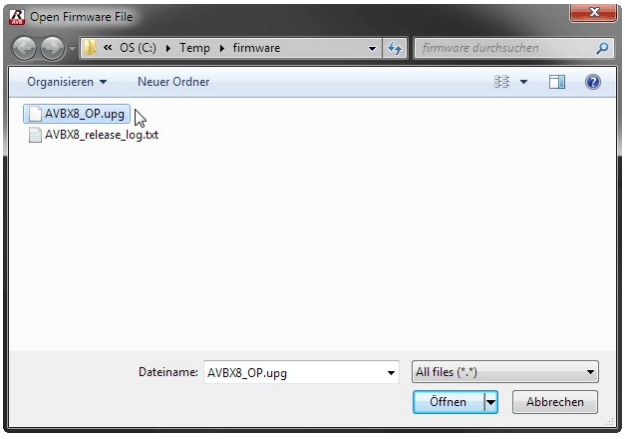


Figure 69: select firmware file

➢ Click on one or multiple (with the *CTRL* and/or *SHIFT* key) entities to select the entities to be updated.

	The firmware file is hardware dependent. Hence, only entities of the same hardware can be updated simultaneously.

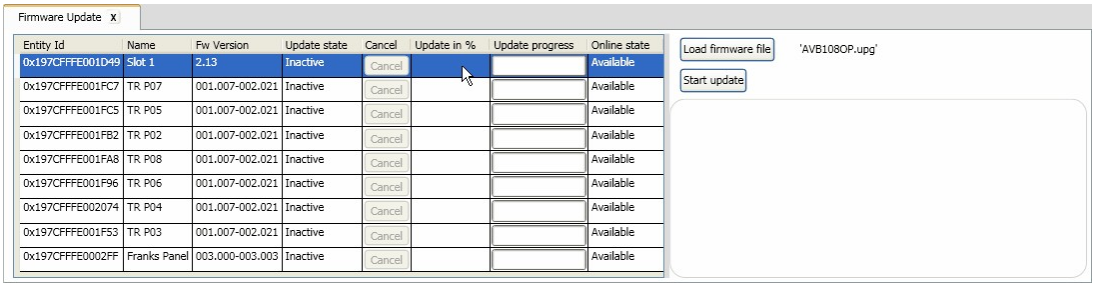


Figure 70: select entities to be updated

➢ Click the button *Start update*.

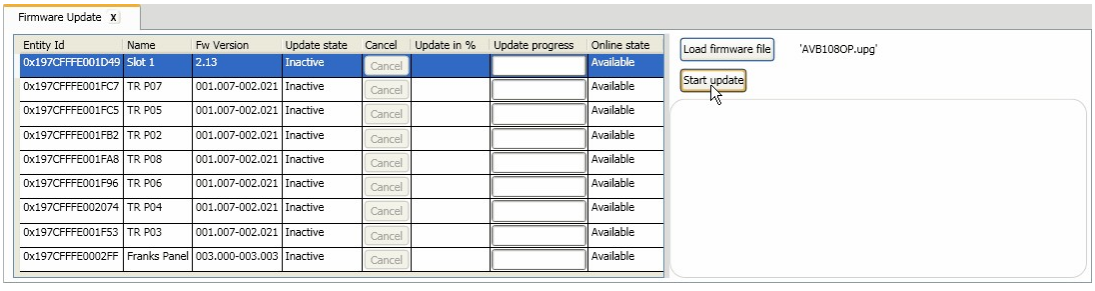


Figure 71: start the update

➢ Watch the update progress.
Depending of the option setting *Reboot entity after Firmware Update per default*, the entities will reboot after completing the update procedure.

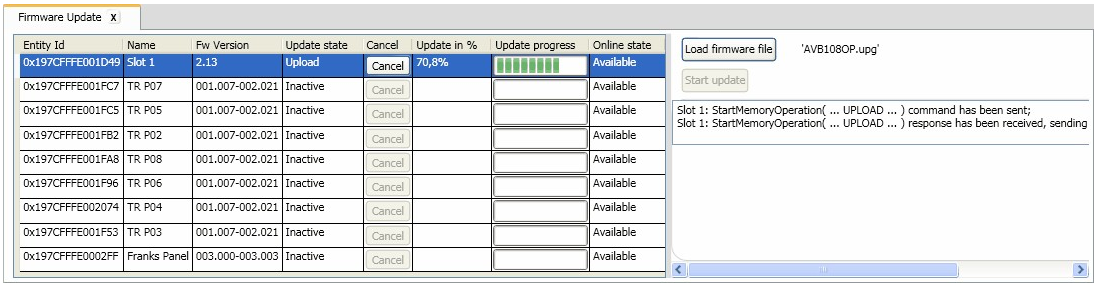


Figure 72: update progress

Only entities containing the three following descriptor values, are available for the firmware update:

ENTITY	Entity capabilities	**EFU_MODE**	
		ADDRESS_ACCESS_SUPPORTED	
CONFIGURATION	MEMORY_OBJECT	Memory object type	**FIRMWARE_IMAGE**

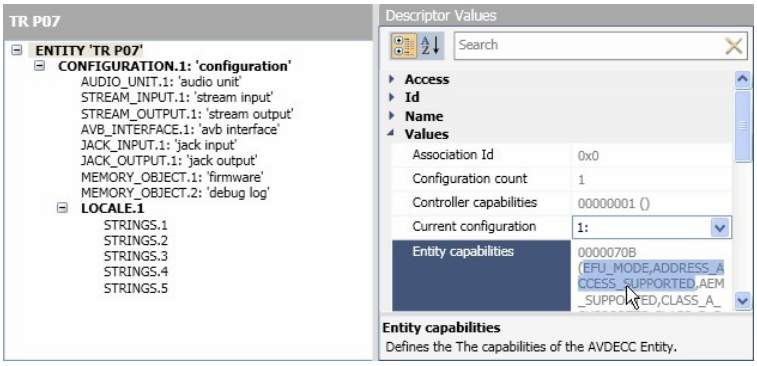


Figure 73: required ENTITY descriptor values for firmware update

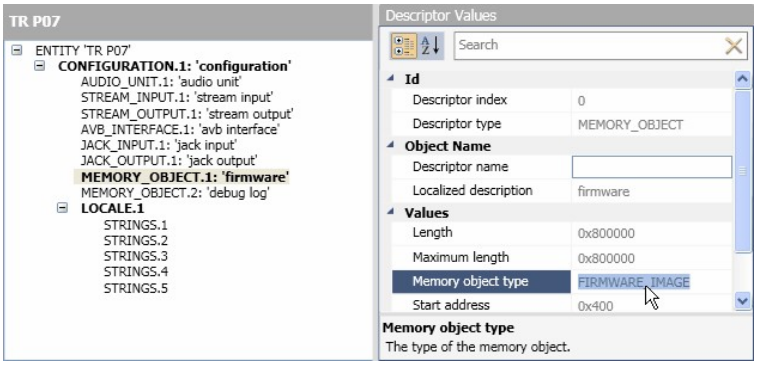


Figure 74: required MEMORY_OBJECT descriptor values for firmware update

5 Appendix

5.1 Glossary

ACMP	**A**VDECC **C**onnection **M**anagement **P**rotocol
ADPDU	**A**VDECC **D**iscovery **P**rotocol **D**ata **U**nit
AEM	AVDECC Entity Model
AES/EBU	Standard for carrying a digital stereo audio signal (2 mono channels) between devices.
AVB	**A**udio **V**ideo **B**ridging
AVDECC	Standard for **A**udio/**V**ideo **D**iscovery, **E**numeration, **C**onnection Management and **C**ontrol
GUID	**G**lobally **U**nique **ID**entifier is a unique reference number used as an identifier
gPTP	**g**eneralized **P**recision **T**ime **P**rotocol
MSRP	**M**ultiple **S**tream **R**eservation **P**rotocol
PC	Personal Computer
Sample Rate	Number of digital amplitude values per second taken from a signal (48/96kHz)

5.2 Service

If you have any further questions, we offer comprehensive customer service options for this product including:

- Telephone Service
- Email Service
- Skype Service
- Fax Service
- Configuration Support
- Trainings
- Repair

Your primary point of contact for any service issues is your local dealer.
In addition, Riedel Customer Service in Wuppertal, Germany is also available to assist you.

Telephone: +49 (0) 202 292 9400
(Monday - Friday, 8am – 5pm, Central European Time)

Fax: +49 (0) 202 292 9419

Skype: riedel.communications.service

Or use the contact form on our website:
www.riedel.net > Company > Riedel Communications > Contact > Wuppertal (Headquarters)

For repairs, please contact your local dealer. Your dealer will be able to help process your repair as fast as possible and/or arrange for the delivery of spare parts.

The address for repairs sent directly to Riedel Communications GmbH is:

Riedel Communications GmbH & Co. KG
- Repairs -
Uellendahler Str. 353
D-42109 Wuppertal
Germany

Please add a completed repair form to all your repairs.
The form can be found at the Riedel website:
www.riedel.net > Company > Services > Support > Contact

Notes