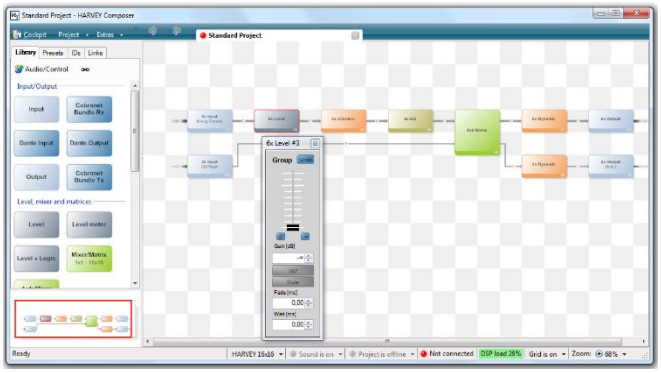
HARVEY®

Operating Manual



HARVEY Composer

Version 3.4

DSPECIALISTS
Digitale Audio- und Messsysteme GmbH
Helmholtzstr. 2-9 L
D-10587 Berlin

www.harvey.audio

© DSPECIALISTS GmbH, Berlin

Table of Contents

1 Introduction 7
1.1 Features of the HARVEY Composer Software 8
1.2 Information about the HARVEY devices 8
1.2.1 Flexible programming 9
1.2.2 Powerful Audio signal processing 9
2 Working with HARVEY Composer 10
2.1 Requirements to Computer 10
2.2 Software Installation 10
2.3 Launching and Exiting the Application 12
2.4 Cockpit 13
2.4.1 Access Control 16
2.5 Main Window 17
2.5.1 Menu Bar 17
2.5.2 Worksheet 17
2.5.3 Sidebar 18
2.5.4 Status Bar 18
2.6 Sidebar 19
2.6.1 Library 20
2.6.2 Presets 21
2.6.3 IDs 22
2.6.4 Links 23
2.7 Offline and Online Modes 26
2.8 Creating Projects 27
2.8.1 Block Properties 27
2.8.2 Connection Editor 28
2.8.3 Port Properties 28
2.8.4 Wiring Function Blocks 29
2.8.5 Context Menu of Function Blocks 30
2.8.6 Context Menu of Worksheet 31
2.8.7 Channel List and Wires 32
2.8.8 Deleting Channels 33
2.8.9 Groups 33
2.8.10 Settings / Function-Block Parameters 34
2.8.11 Copy & Paste of Parameters 35
2.8.12 Undo and Redo 36

2.9 Working with Hypermatrix Projects 37
2.9.1 Overview of Hypermatrix Concept 37
2.9.2 Prerequisites for a Hypermatrix Installation 38
2.9.3 Creating Hypermatrix Projects 38
2.9.4 Programming and Configuration Hypermatrix Projects 39
2.9.5 Transferring Hypermatrix Projects from and to HARVEY Devices 41
2.9.6 Presets in Hypermatrix Projects 41
2.9.7 Troubleshooting 42

3 Audio Function Blocks 44

3.1 Audio Function Blocks: Inputs and Outputs 45
3.1.1 Input (Audio Function Block: Input / Output) 46
3.1.2 Output (Audio Function Block: Input / Output) 47
3.1.3 CobraNet Rx (Audio Function Block: Input / Output) 49
3.1.4 CobraNet Tx (Audio Function Block: I/O) 50
3.1.5 Dante Input/Output (Audio Function Blocks: Input / Output) 51
3.1.6 AES3 Input/Output (Audio Function Blocks: Input / Output) 52
3.2 Audio Function Blocks: Level, Mixer, and Matrices 53
3.2.1 Level (Audio Function Block: Level, Mixer, and Matrices) 54
3.2.2 Level Meter (Audio Function Block: Level, Mixer, and Matrices) 55
3.2.3 Level » Logic (Audio Function Block: Level, Mixer, and Matrices) 56
3.2.4 Mixer/Matrix (Audio Function Block: Level, Mixer, and Matrices) 57
3.2.5 AutoMixer (Audio Function Block: Level, Mixer, and Matrices) 59
3.2.6 Input Selector (Audio Function Block: Level, Mixer, and Matrices) 61
3.3 Audio Function Blocks: Filter and EQ 63
3.3.1 1..8-Band EQ (Audio Function Block: Filter and EQ) 64
3.3.2 Filter (Audio Function Block: Filter and EQ) 66
3.3.3 Crossover (2..4) (Audio Function Block: Filter and EQ) 68
3.3.4 AEC – Acoustic Echo Cancellation 70
3.4 Audio Function Blocks: Dynamic 74
3.4.1 Dynamic (Audio Function Block: Dynamic) 75
3.4.2 AGC (Audio Function Block: Dynamic) 77
3.4.3 Ducker (Audio Function Block: Dynamic) 79
3.4.4 AVC (Audio Function Block: Dynamic) 81
3.4.4.1 Guide to Using the AVC Block 82
3.5 Audio Function Blocks: Function Blocks 85
3.5.1 Delay (Audio Function Block: Function) 86
3.5.2 a – b Differential Signal (Audio Function Block: Function) 87

3.5.3 Sine/Noise Generator (Audio Function Block: Function) 88

4 Control Function Blocks 89

4.1 Control Function Blocks: Inputs and Outputs 89
4.1.1 Logic Input (Control Function Block: Input / Output) 90
4.1.2 Logic Output (Control Function Block: Input / Output) 91
4.1.3 Voltage Input (Control Function Block: Input / Output) 92
4.1.4 Wind » Level 93
4.2 Control Function Blocks: Serial Interfaces 95
4.2.1 RS-232 Interface (Control Function Block: Serial Interfaces) 96
4.2.2 RS-485 Interface (Control Function Block: Serial Interfaces) 97
4.2.3 TCP Server (Control Function Block: Serial Interfaces) 98
4.2.4 UDP Client (Control Function Block: Serial Interfaces) 99
4.3 Control Function Blocks: Logic 100
4.3.1 Negator (Control Function Block: Logic) 101
4.3.2 RS Flipflop (Control Function Block: Logic) 102
4.3.3 T-Flipflop (Control Function Block: Logic) 103
4.3.4 AND (Control Function Block - Logic) 104
4.3.5 OR (Control Function Block - Logic) 105
4.3.6 XOR (Control Function Block - Logic) 106
4.4 Control Function Blocks: Function Blocks 108
4.4.1 Preset (Control Function Block: Function Blocks) 109
4.4.2 Level Up/Down 110
4.4.3 Zone Control 111
4.5 Control Function Blocks: Serial 114
4.5.1 Serial » Logic (Control Function Block: Serial) 115
4.5.2 Logic » Serial (Control Function Block: Serial) 116
4.6 Control Function Blocks: DMX 117
4.6.1 DMX (Control Function Block: DMX) 118
4.6.2 DMX » Gain Control (Control Function Block: DMX) 120
4.6.3 DMX » Logic (Control Function Block: DMX) 121
4.7 Control Function Blocks / Controls 122
4.7.1 Web-Interface (Control Function Blocks / Controls) 123
4.7.2 RC4/RC12 Remote Control – Introduction 126
4.7.3 RC4 (Control Function Blocks / Controls) 129
4.7.4 RC4-DMX (Control Function Blocks / Controls) 131
4.7.5 RC12 (Control Function Blocks / Controls) 133

5 Controlling and Integrating Media Systems 135

5.1 Interface-Format Conversion 136
5.2 Integrating Proprietary Protocols 137

6 HARVEY Device Variants 139

1 Introduction

HARVEY is a family of freely programmable audio and media control devices that are ideal for conference rooms, theaters, museums, home theaters, educational institutions or multi-purpose halls, without being limited to these applications.

With a wide range of audio and control interfaces, HARVEY devices can connect to a wide range of devices and act as an intelligent control center for audio, lighting and media technology.

The integrated powerful digital signal processors (DSP) with their extensive software capabilities allow even for complex systems with a high number of channels a tailor-made and high-quality audio processing in real time with minimal signal delay of less than a millisecond.

To control HARVEY and the devices connected to it all common media controls of well-known manufacturers are suitable, such as Crestron, AMX, Cue and Apple, by using the integrated text protocol “H-Text”.

HARVEY devices also have the special ability to process and generate foreign protocols that can be programmed by you with almost no effort. In addition, they even allow data to be transferred between their serial interfaces (Ethernet, RS485, RS232), thus rendering additional converters superfluous.

Thanks to the integrated web server, graphical user terminals can be implemented quickly and easily for the end user.

All settings can be aggregated into scenarios, saved as a preset and recalled at the push of a button to quickly switch between different usage options of the installation.

The wall-mounted "HARVEY RC4" network-based remote control allows control of presets, audio levels and DMX lamps.

Both the HARVEY devices and the software have a modular structure. The programming of all HARVEY components is done conveniently via Windows computer with the freely accessible software "HARVEY Composer".

By programming with the HARVEY Composer software offline - even without connected devices - new installations and changes can be planned and prepared from any location. The programmed project is transferred to the hardware via an IP network and fine-tuned on the device online in real time in the HARVEY Composer software.

In order to exploit all possibilities of the HARVEY family, we recommend that you read this HARVEY Composer manual carefully.

With the HARVEY H-Text protocol, all parameters of HARVEY can be remotely controlled via serial interfaces (e.g. via WLAN via TCP). An additional manual explains in detail how the protocol works.

Further manuals are available for the different devices of the HARVEY family.

1.1 Features of the HARVEY Composer Software

HARVEY Composer is a Windows computer program that allows you to conveniently program HARVEY devices as an integrator of PA systems and conferencing systems.

The team behind the HARVEY Composer has set itself the goal of creating an intuitive and time-saving programming environment:

- ➔ HARVEY Composer is the central programming environment for all HARVEY device variants: The specific device properties resulting from their features are managed by the HARVEY Composer.
- ➔ Signal processing and control functions are organized as graphically displayed blocks in an audio and control plane.
- ➔ Function blocks are taken from a library area using the drag-and-drop principle and freely placed and connected on a worksheet.
- ➔ Equivalent connections between function blocks are grouped for a better overview. Nevertheless, details in the representation of individual connections are not lost.
- ➔ It is possible to work without a device connection (offline) - a resource display also always shows offline the current processor load for the preselected device variant.
- ➔ When the HARVEY Composer is connected to a HARVEY device (online), all parameter changes (volume, filter settings, etc.) are performed in real time.
- ➔ Device access can be protected against unauthorized access with a password.
- ➔ The connection to the HARVEY devices takes place via Ethernet network, without requiring special IT skills: All HARVEY devices that can be reached in the network are found via their changeable text designations and offered for selection.
- ➔ HARVEY Composer together with compatible HARVEY devices integrate the so called Hypermatrix, a powerful technology which enables you to design a multi-device system in a single HARVEY Composer worksheet. This allows both the interfaces and the processing power in a HARVEY system to be scaled. Additionally a distributed HARVEY system becomes easily manageable.

1.2 Information about the HARVEY devices

The HARVEY product family is constantly expanding and currently consists of the following members:

- ➔ HARVEY mx.16 [Dante] [Cobranet]: DSP audio and control matrix - fixed 16x16 analog input / output channels with / without Dante or Cobranet capability in two height units.
- ➔ HARVEY NxM [Dante][Amp][AES]: DSP audio and control matrix - flexible equipment with a selectable analogue input (N) and output channel count (M) with up to a sum of 32 channels, with / without Dante capability (64x64 channels), with / without amplifiers (4, 8 or 12 channels) and with / without AES3 digital audio interfaces in one height unit.
- ➔ HARVEY RC4: Power over Ethernet (PoE) powered control module with buttons, one rotary knob and RGB signaling for wall, furniture or rack mounting.

1.2.1 Flexible programming

All HARVEY devices have a comprehensive set of features that can handle most applications in audio and conferencing systems.

At the same time, there are no rigid processing structures:

- ➔ As a user, you specify the sequence in which the audio signals are to be processed and combined.
- ➔ The control functions range from freely definable state changes through buttons or switches connected to the HARVEY devices, via the possibility of converting interface formats (e.g. from Ethernet to RS232) to a freely configurable web interface for a time-saving creation of end-user terminals.

1.2.2 Powerful Audio signal processing

Thanks to their powerful 32/40 bit signal processors, the HARVEY audio and control matrices have enough processing power to process a large number of channels in real time:

- ➔ For example, a device processes more than 110 channels of 8-band equalizers or 240 seconds of single-block delays or 512 seconds of node delays of a mixing matrix with integrated nodal gain and delay.
- ➔ Depending on the device type, processing and signal conversion takes place with a very small signal delay of 0.75 milliseconds (analog input to output with HARVEY Pro). This delay, also called signal latency, is independent of the number of signal processing or control functions used.

  The following processing blocks add delay to an audio signal depending on its block parameters:

  - Filters (phase distortion of recursive filters)
  - Dynamic blocks (look-ahead time)
  - Delay and mixing matrix with delay (delay value)

- ➔ Signal processing works in floating-point format, so you do not have to worry about the quality of a signal going through multiple gain steps - lowering it by 60 dB, for example, and then catching up through amplification will not lead to quality degradation.

2 Working with HARVEY Composer

2.1 Requirements to Computer

The HARVEY Composer software requires a computer with the following minimum requirements:

Operating System	Windows 7 (32/64 Bit) or Windows 10 (32/64 Bit)
CPU / Processor:	1.5 GHz processor or faster (multi-core recommended)
Memory:	2 GB RAM or more (3 GB recommended for Windows 7 and Windows 10)
Graphic Card:	1024 x 768 resolution (higher recommended)
Network Interface:	100 Mbit/s or 1000 Mbit/s Ethernet Network

2.2 Software Installation

Please make sure you have the latest version of the HARVEY Composer software.

➔ The current version of the HARVEY Composer software can be found on the website www.harvey.audio

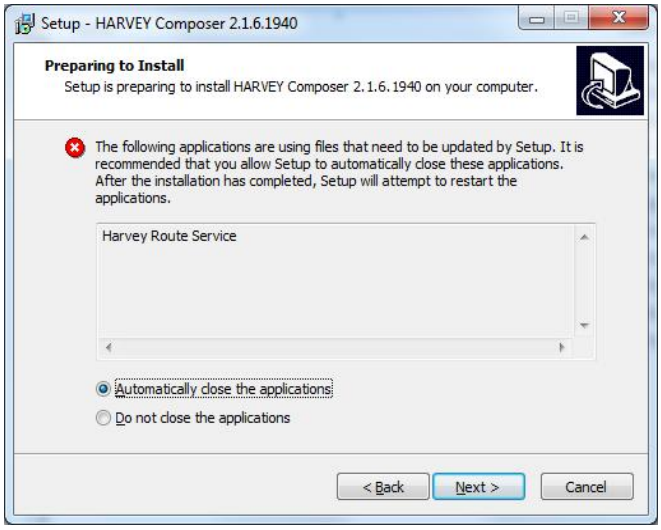
➔ Windows 10: Activate the Microsoft .NET Framework 4.0 under "Control Panel→ Programs→Activate / deactivate Windows Features"

Installation

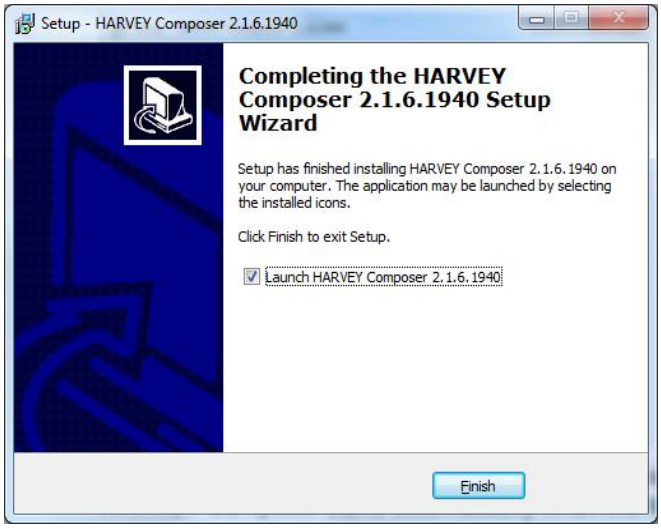
1. Run the installation file "HARVEY Composer-ABC-setup.exe".
2. Depending on the operating system used, Windows may follow up with security warnings. These alerts are intended to protect users from harmful software from the Internet. Files downloaded from our website pose no threat: alerts can be safely confirmed.
   Confirm any warning messages from the operating system with [Execute] or [Yes].
3. Select the language to be used during the installation.

4. In the following welcome window, confirm with [Next].
5. Select the destination folder for the program installation and confirm with [Next].
6. Assign a name for the Windows Start menu entry. The default is "HARVEY". Confirm with [Next].
7. Determine whether additional icon launch shortcuts should be created on the desktop or in the Windows Quick Launch. Confirm with [Next].
8. The following is a list of the selected installation parameters. Confirm with [Next] to start the installation.

9. If HARVEY Composer has already been installed, the installation will report that additional applications necessary for the HARVEY Composer function must be closed. It is recommended that the applications be closed automatically.



10. The software is now completely installed. Click [Finish] in the last dialog window. HARVEY Composer starts automatically when the tick is set in the appropriate place.



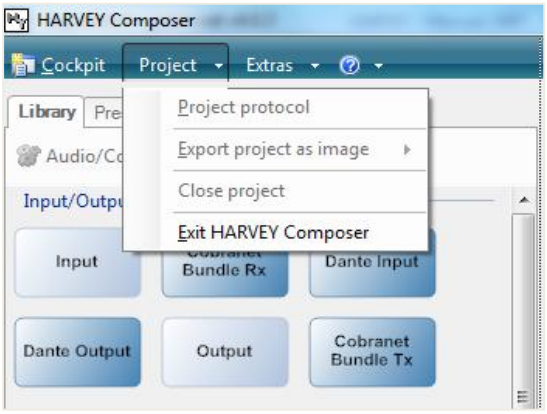
2.3 Launching and Exiting the Application

Launch HARVEY Composer by clicking the program icon:

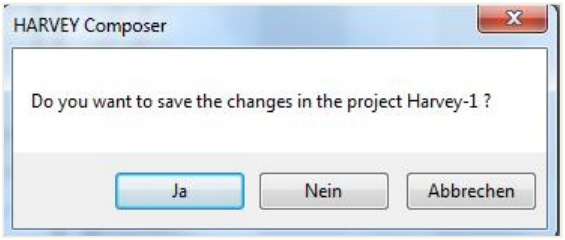
This program icon exists in the Start > All Programs > HARVEY program group (or whichever name you have assigned during installation).

You can also launch HARVEY Composer from the Quick Launch bar or by double-clicking the appropriate desktop icon. In Windows 10, start the HARVEY Composer via the program icon in the Tile view.

For exiting select the HARVEY Composer menu entry [Project > Exit HARVEY Composer] in the main window.



If there are unsaved changes in open projects, HARVEY Composer will prompt you to save them as necessary:



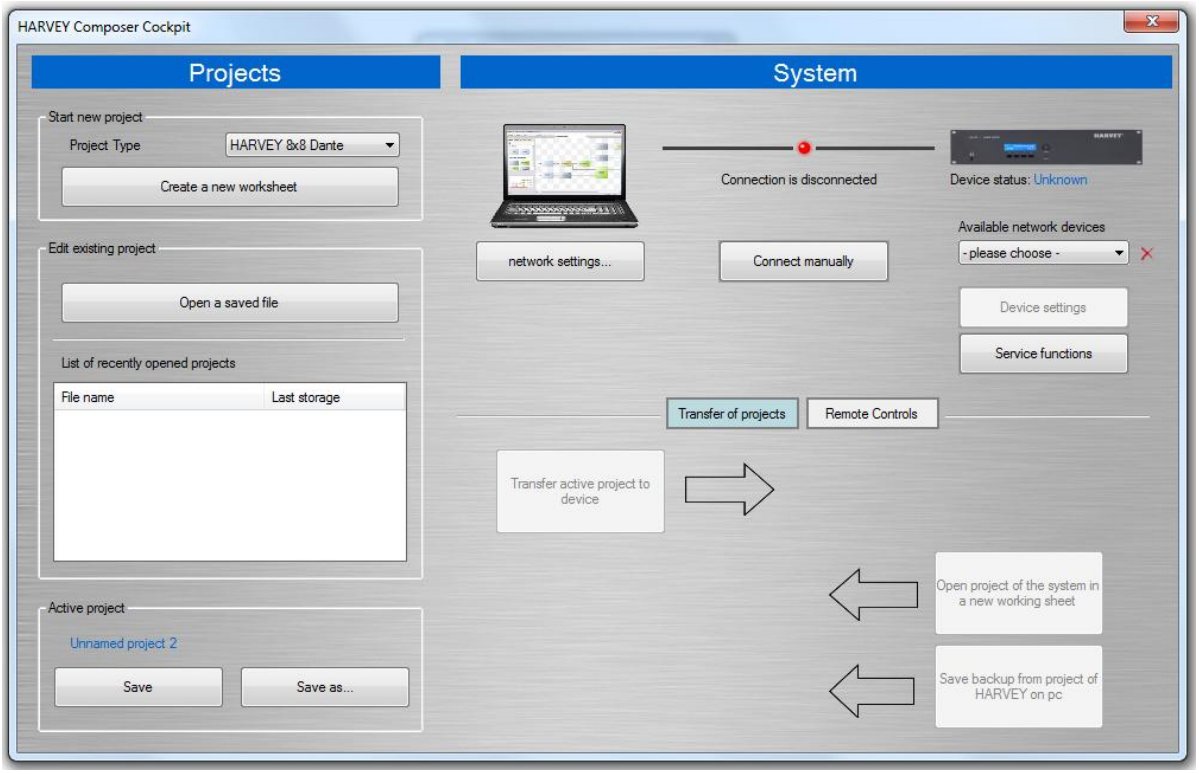
Click [Yes] to store the project. Before the project is stored and the program closed, you have the option of entering comments into a separate window.

If you want to exit the program without saving your changes, click [No].

If you rather do not want to exit the program at this time, click [Cancel].

2.4 Cockpit

At program launch, HARVEY Composer always shows the cockpit. The cockpit is the control center of the software and is subdivided into two sections.



On the left-hand side, you see the **<Project>** section, which allows for creating a new worksheet.

- ➔ Select a **[Project Type]**. The project type determines for which device variant the composer project is to be created. Basically, there are two project types available:
  1. A single device project: Choose a device variant suitable for the intended application. The variant can be changed subsequently in the status bar of the main window (page 17). Such a project cannot be changed to a multi-device project (Hypermatrix) subsequently.
  2. A multi-device project: Choose Hypermatrix which allows you to design a project for up to ten devices in one worksheet (see page 37). This type cannot be changed subsequently, but the device variants being part of the Hypermatrix project can be changed subsequently.
- ➔ Click the **[Create a new worksheet]** button to create a new project.
- ➔ Projects are created offline, meaning that you don't need a live network connection from the computer to a HARVEY unit; you need such a connection only when you want to transfer the finished project to a HARVEY unit.

Already saved projects can be opened in the next section. A list shows all recently opened projects. The currently active project is shown at the bottom and can be saved using the **[Save]** and **[Save as]** buttons.

The **System** area on the right-hand side provides access to the network settings of your computer and allows for establishing connections to the HARVEY units on your network. All HARVEY units are automatically detected and listed in the **<Available Network Devices>** drop-down menu.

➔ To establish a connection, select the desired HARVEY system and click on "Connect". The red light then turns yellow, and when the connection is successful, it changes to green and the text "Connection is established" appears.

Under **[Device Settings]**, the device name and network settings can be managed, the access control for the device can be configured (see “Access Control” on page 16), the device can be restarted or muted ("Mute").

➔ The device name allows you to find specific HARVEY devices on the network (see **<Available Network Devices>** above) easily.

➔ **Important to know:** Each HARVEY device transfers its name to the local Dante endpoint (i.e. device name displayed in Dante Controller). That means:

- After a change of the HARVEY device name the Dante endpoint name is automatically renamed.
- If, before renaming, you have already set Dante routes, you have to reconfigure the routes with the help of Dante Controller after renaming.

The **[Service Functions]** button provides access to functions used for updating the HARVEY unit firmware and the HARVEY Composer software. Furthermore, version information about the connected device is displayed in the area "Service functions". A button **[Open log directory]** opens the directory in the file explorer where HARVEY Composer stores backup copies of the Composer projects and log files that are useful for analysis in case of an error.

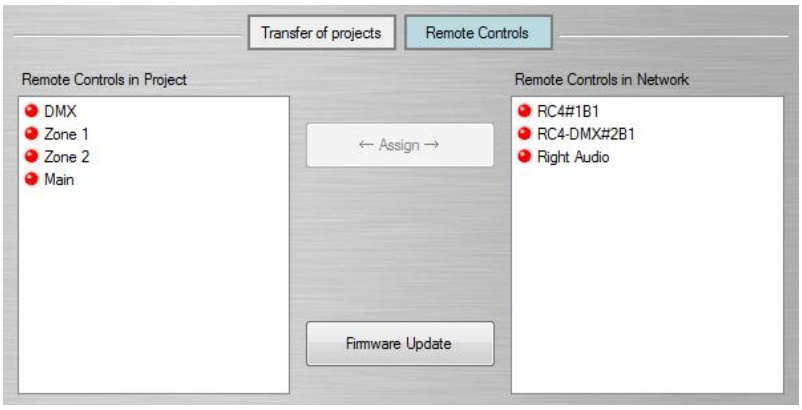
The lower section is used for **[Transfer of projects]**. If the HARVEY Composer is connected to a HARVEY device, a finished project can be transferred to the device (see also changing the state offline to online in the status bar of the main window on page 16), the project stored in the HARVEY device is loaded into the Composer or saved as a backup on the PC without displaying it.

➔ If the project type differs from the device variant, HARVEY Composer will try to convert the project to the connected device variant when transferring to the device - depending on the difference, there are two cases:

1. Convertible: A message appears that the project can be converted. Confirm with [OK] or end with [Cancel].
2. Non-convertible: A message appears stating that the project cannot be converted and the incompatible functions are displayed.

➔ After successfully transferring a project to a device, the project status is online (see section "Offline and online mode" on page 26).

The button **[Remote Controls]** opens an area allowing HARVEY RC4 control modules to be assigned to the active Composer project:



➔ Read in section RC4/RC12 Remote Control – Introduction for instructions on using the RC4 control modules.

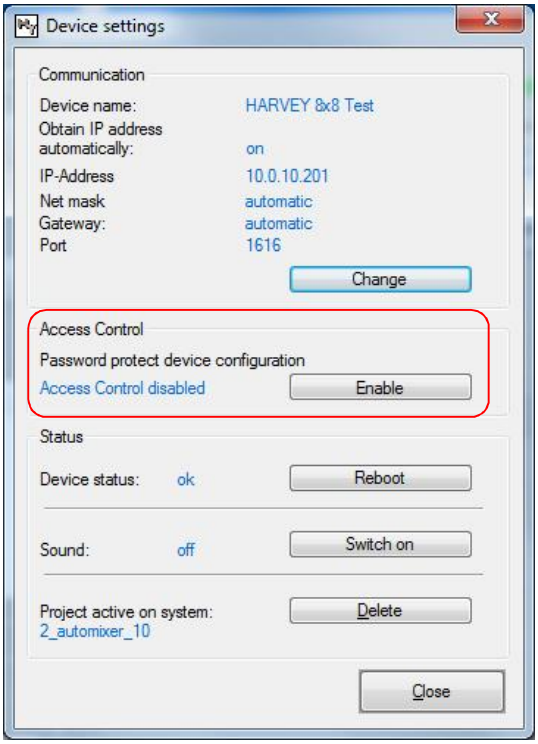
2.4.1 Access Control

With the HARVEY Composer, the project stored on the device can be protected against unauthorized access with a password. The following actions are only possible with activated access protection after entering the valid password:

- ➔ Transfer a project from the HARVEY Composer to the device
- ➔ Load a project from the device into a new worksheet of the HARVEY Composer
- ➔ Creating a backup copy of a project that is located on the device
- ➔ Turning the Mute state of the device on / off ("Tone On / Off")
- ➔ Perform a firmware update
- ➔ Cancellation of access protection

These restrictions do not allow an unauthorized user to change the project configuration or individual parameters of the project.

To enable access protection, first open the Composer Cockpit, connect to the device you want, and then go to **[Device Settings]**.



Under the entry "Access Control", the device reports the state of the access control either with the output "Protection is deactivated" or "Protection is activated".

- ➔ If there is firmware on the device that does not yet contain the "access control" feature, the status "not available" appears.

After pressing the **[Enable]** button, a dialog box for entering a password appears, which must be entered twice before the confirmation button **[Activate Access Control]** is activated.

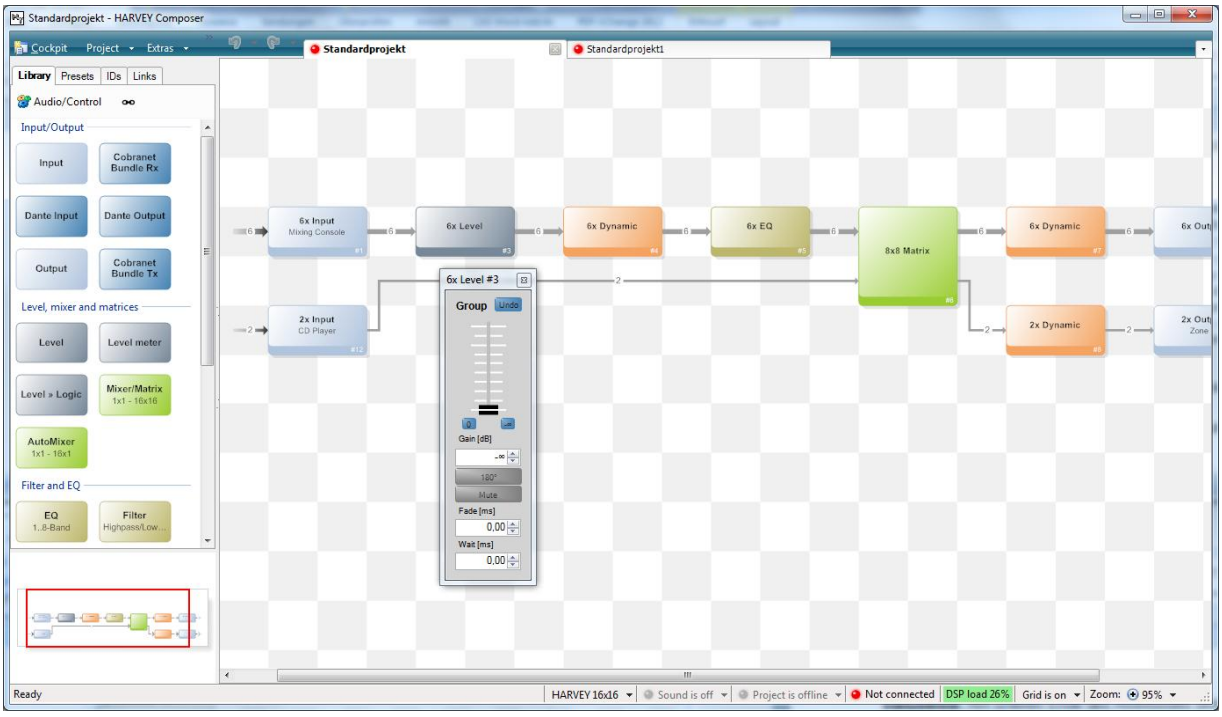
- ➔ Make a note of the password, because if the password is lost the access protection can only be disabled with the help of the manufacturer.

- ➔ The access control is stored on the device.
- ➔ The password to unlock access control needs to be entered only once per composer session, so after activating the access protection, without reconnecting to the device, all protected actions can take place without repeated password entry.

Each time you reconnect to a protected device using HARVEY Composer and attempt to perform one of the protected actions, you will be prompted once for password entry.

2.5 Main Window

The user interface of the HARVEY Composer is divided into different areas.



2.5.1 Menu Bar

In the menu bar you can call the **[Cockpit]**. Under the menu item **[Project]** the sub-window for the Hypermatrix Device Setup can be invoked, the project protocol is found, the current configuration can be exported as an image in the formats .emf, .bmp, .jpg or .png, the current project can be closed or the program can be terminated. Under **[Extras]** you will find the language options and you can choose between German, English and Chinese. Additionally there you can find user-specific settings of the HARVEY Composer. Here you can set the **relative reference level** <dBr> in the project (default value: 18 dBr = -4 dBFS).

➔ The set reference level <dBr> does not affect thresholds or levels of audio signals, and is only used to scale signal levels.

2.5.2 Worksheet

In the right pane, called Worksheet, the HARVEY project is created and configured. A project refers to the complete configuration and parameterization of a HARVEY device. All functions are displayed with blocks on the worksheet. Function blocks can be configured by double-clicking and are connected to wires on the worksheet.

➔ Using project tabs above the worksheet, you can switch between different projects. This allows multiple projects to be edited independently of each other.

➔ Use the mouse wheel to zoom in and out of the worksheet.

2.5.3 Sidebar

The left pane of the main window displays the **Library** (with all function blocks) and provides access to the **Preset**, **IDs** and **Links** menu. In the lower section is the "Bird's View" window.

- ➔ The sidebar provides all tools for project building and also serves as a management center.
- ➔ See page 19 and following for a detailed functional description of the sidebar.

2.5.4 Status Bar

At the bottom far right of the worksheet is **[Zoom]**: 5 selectable zoom levels change the view size on the worksheet.

The **[Grid]** button activates or deactivates the automatic grid arrangement. The grid helps you enforce clean project structures as the blocks snap into place on the worksheet. If the grid is deactivated, blocks can be freely placed.

The **<DSP Utilization>** display shows the percentage utilization of the DSP, even in offline mode.

In addition, you can see from the status bar whether the project is being edited **[online]** or **[offline]** (see section “Offline and Online Modes” on page 26). A mouse click on this field changes the state from offline to online or vice versa. The field also works as a drop-down menu in which the target state can be selected. After selecting <online>, the current project is transferred to the HARVEY device (see also "Transfer of projects” in the Cockpit on page 13).

- ➔ If the project type differs from the device variant, HARVEY Composer will try to convert the project for the connected device variant - depending on the difference, there are two cases:
  1. Convertible: A message appears that the project can be converted. Confirm with [OK] or end with [Cancel].
  2. Non-convertible: A message appears stating that the project cannot be converted and the incompatible functions are displayed.

It also indicates if and with which **<HARVEY device>** a project is connected.

Finally, a box indicates if the **[Sound]** is on.

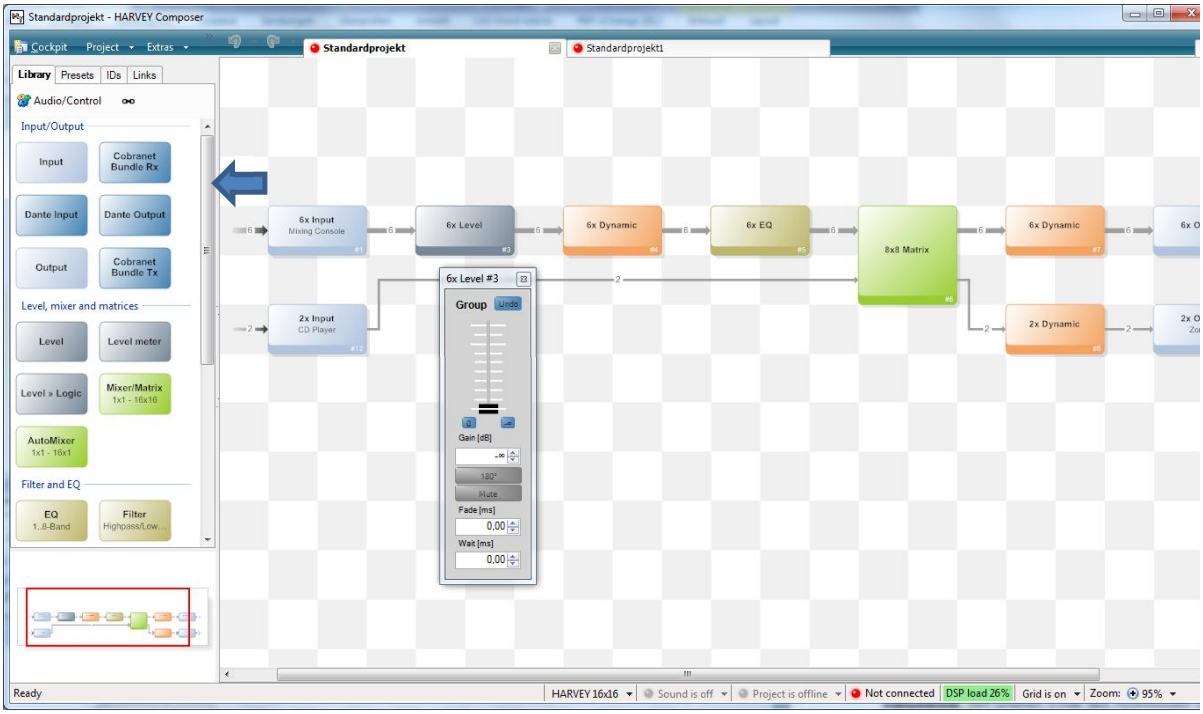
- ➔ Sound is switched off: All audio outputs of the HARVEY device are muted.
- ➔ **Sound is always switched off during a project transfer, and must be switched on manually after project transfer.**

On the far left, the **[Project Type]** can be changed in offline mode.

- ➔ The associated drop-down menu shows only those device variants to which the project is compatible.
- ➔ If the project contains structural differences - for example, if there are more audio inputs than a specific device type, then that device type will not be displayed.
- ➔ If only selected parameters in the project are different from those of a device type, then HARVEY Composer converts the parameter(s) to the next possible value according to the new device type.

2.6 Sidebar

In the sidebar, in the left work area of the main window, you can call various menus via tabs: **<Library>**, **<Presets>**, **<IDs>**, and **<Links>**.



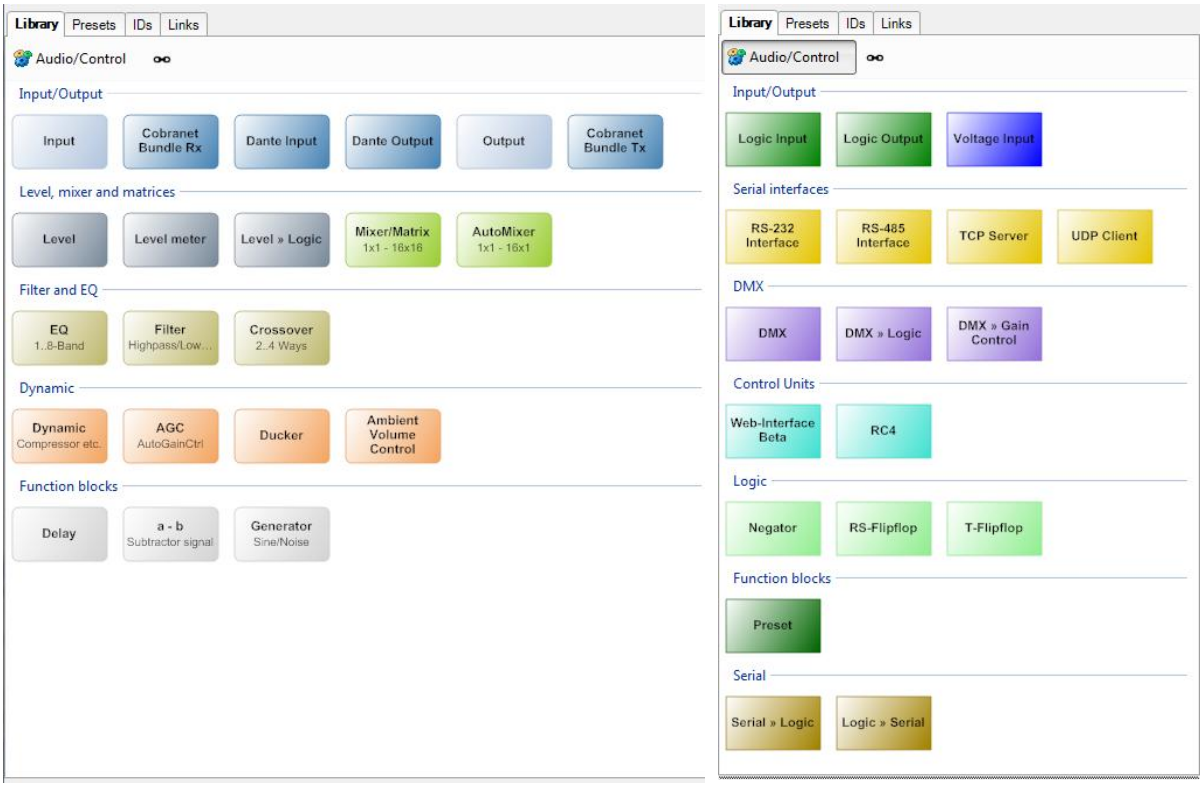
Bird's View: The Bird's View is located in the bottom left corner. It provides a full overview of your project. In this view, the area of the worksheet that is currently visible on the right-hand side is marked with a rectangular frame that can be shifted using the mouse.

➔ You can expand and collapse the sidebar using the icon with the box containing the two arrows on the right-hand border.

➔ To change the height of the Bird's Eye view, use the sidebar splitter.

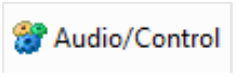
2.6.1 Library

The **Library** contains all function blocks. Use the drag & drop functionality to move function blocks to the worksheet. There are two function-block layers: “Audio” and “Control”.



Blocks in the **Audio** category include analog and digital audio inputs and outputs, matrices, mixers, meters, level faders, filters, equalizers, dynamic processors, and more.

In the **Control** category analog, digital and serial control interfaces, such as logic inputs and outputs, voltage inputs, RS232, RS485, DMX, Ethernet and logic function blocks are available.



toggles between the two layers <Audio> and <Control>.

At the same time, the appearance of the function blocks and connections on the worksheet is adjusted.

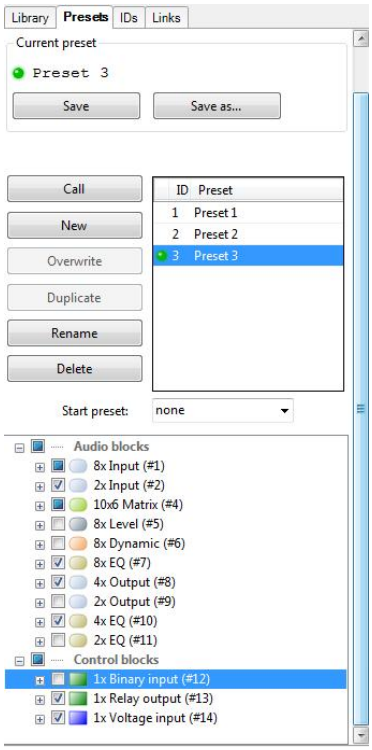
Function blocks on the worksheet and their connections are visually focused on audio signal processing or control signal processing to make the project more manageable. All functional blocks that cannot be connected in the active layer are displayed as transparent blocks. Function blocks that provide connection types for both audio and control signals switch to the other graphic layer and show the corresponding connection options.

The chaining icon displays the appearance of "Links" on the worksheet. More about "Links" can be found on page 23.

➔ If function blocks are incompatible with the set project type (see sections "Cockpit" page 13 and "Main window" page 16), a message will appear when dragging and dropping such a block and the placement of this block will be aborted.

2.6.2 Presets

In the Presets menu in the sidebar, you create new presets and recall and manage existing ones. A preset is a snapshot of a parameter set. For example, a preset can include the current parameter settings of all function blocks as well as a single parameter of just one function-block channel.



Each preset has a unique ID. You can recall presets from the “Preset” function block or using H-Net and H-Text messages. Note that when recalling a preset, settings contained in it will overwrite current parameter settings. Since a HARVEY device can always hold just one project at a time, presets are the key to utilizing the full potential of the HARVEY system’s flexibility.

All function blocks that are part of the current project are displayed in a tree near the window bottom. Clicking the [+] signs allows for showing or hiding additional tree levels, so you can select individual function-block channels and even their parameters.

To create a new preset, first select the parameters to be stored:

- ➔ Enable the checkboxes of all function blocks, individual channels, and parameters to be included in the preset.
- ➔ Click the [Save] or [Save As] buttons and name your project.

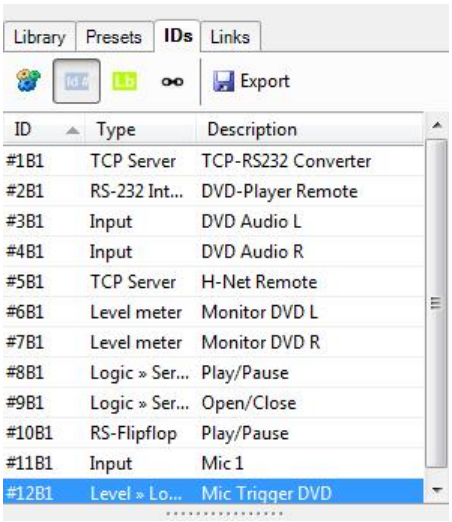
A list of existing presets will be displayed where you can click to select individual presets for recall, overwriting, duplicating, renaming, or deleting.

From the **<Start Preset>** drop-down menu, you can select a default preset that is automatically loaded to the HARVEY device at boot time. If you do not configure a default preset, all settings that were configured before the last shutdown will be recalled from the persistent memory.

- ➔ If you are going to configure various system setups, be sure to define a start preset first. This way, you avoid unpredictable and mixed parameter settings at boot time.
- ➔ Attention: When restarting a single device in a hypermatrix system, only the block parameters assigned to this device are loaded from the start preset. Start presets defined for all devices are therefore only fully implemented when all devices are restarted. See 2.9.6 Presets in Hypermatrix Projects.

2.6.3 IDs

The IDs menu located on the sidebar lists all channel IDs.



Each function block and each channel contained in it has a unique ID within the project.

An ID is composed as follows:

 # <function_block_ID> ***B*** *<channel ID>*

Example: **#3B2** (function block 3, channel 2)

The <ID> column lists all IDs in ascending order.

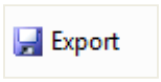
The <Type> column shows the type of function block.

To assign a custom label to a channel, double-click the corresponding item in the <Description> column and enter the label.

Clicking this button toggles between the <Audio> and <Control> layers on the worksheet.

Use these buttons to select whether the channel list of a function block shows the **ID** or the custom **La**b**e**l assigned in the <Description> column.

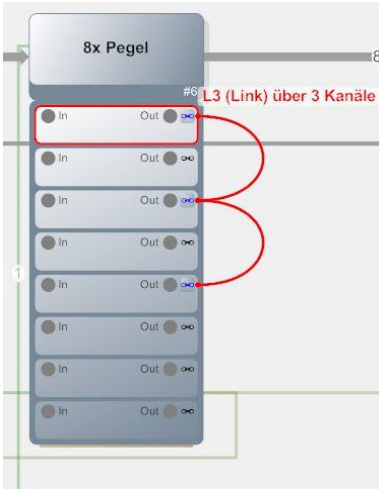
Using the chain icon , you can toggle between showing and hiding link groups on the worksheet. (For more information on link groups, see page 23.)



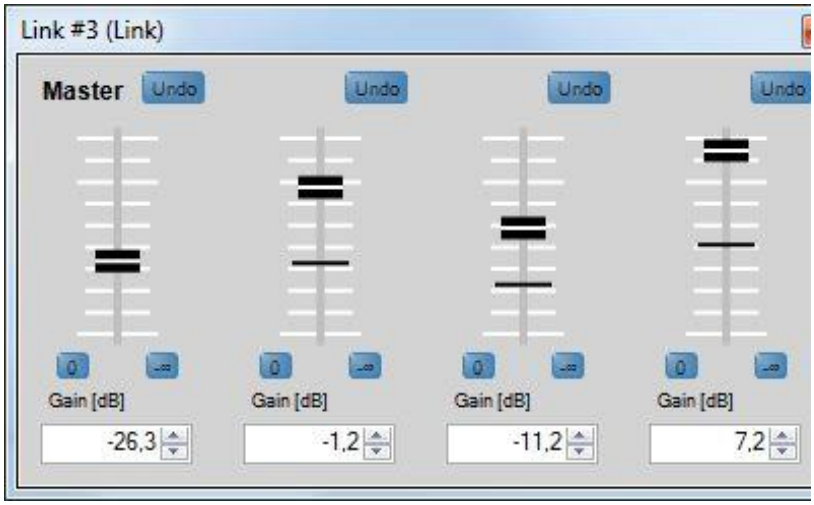
The export function allows for writing the entire channel list including the IDs, the function-block type, and the description to a CSV file.

2.6.4 Links

The Link function provides for linking channels of level function blocks. When level faders are linked, their current level offsets will be maintained, and you can control all of them using a single “master” fader. When you create a new link group, its master fader will default to -∞ dB, meaning that all channel faders are effectively set to –∞ dB. Setting the master fader to unity gain (0 dB) will restore the original offsets of the channel faders. The dB setting of the master fader will be added to the settings of all link-group members (offset).



(Channel view with Link)



(Link settings)

➔ You can access the link functions from the **<Links>** menu in the sidebar.

➔ You can even link channels from multiple “Level” blocks.

To set up a new link group or to ungroup channels, first select the channels in a block’s channel list.

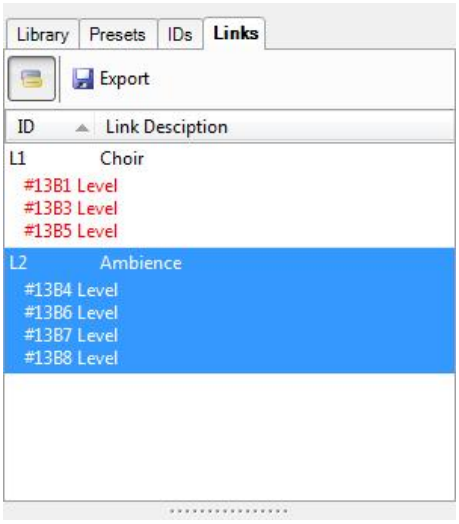
➔ While holding the <Ctrl> key on your keyboard, click each channel you want to add to or remove from the link group. When you’re finished, release the <Ctrl> key. All selected channels will now have a red border.

- ➔ Next, to create the link group, point to one of the selected channels. Right-click the selected channel and select **<Create New Link>** from the pop-up menu. In the Link Properties window, you can configure a name for the link group.
- ➔ If you want to ungroup all channels of a link group, right-click any member channel of the group and select **<Release Current Link>**.
- ➔ To remove one or more channels from a link group, select the channels to be removed, then right-click and select **<Release From Current Link>** from the pop-up menu.
- ➔ You can also select a contiguous range of channels (for example, channels 2-5). To do so, click the first channel, then press and hold the <Shift> key while clicking the last one.

Link groups are managed in the **<Link Settings>** dialog.

- ➔ To access that dialog, right-click on the channel list and select **<Link Settings>**, or double-click on the channel ID in the Links menu in the sidebar. Alternatively, you can double-click any of the red connecting lines of a link next to the channel list to access the link settings.

The master fader is always located on the left-hand side of the Link Settings window. To the right of the master fader, you can see the channel faders of the link members, each displaying the original fader setting. The horizontal lines show the current level settings of each channel based on the offset master-fader setting. Link members are marked on their function blocks with red connecting lines, the link ID, and the assigned labels.



The Links menu in the sidebar lists all configured link groups with their respective names.

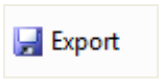
Each link group has a unique <ID> consisting of the letter “L” and an automatically assigned number.

The <Link Description> shows the assigned name.

- ➔ If the <Link> tab has been selected in the sidebar, the worksheet shows the linked channels. All components that do not belong to any link-group member are grayed out on the worksheet.

➔ The <Library> and <IDs> sidebar menus provide a chain button for toggling the view.

Using the <Show/Hide Blocks> button, you can set whether the relevant function blocks are listed below the link ID or not. Member channels from any blocks are listed with their channel ID.



The export function allows for writing the entire list including all details to a CSV file.

2.7 Offline and Online Modes

In HARVEY Composer, you will use the offline mode for creating new projects.

You can configure, arrange, link, and wire function blocks on the worksheet in offline mode only. When you have established a connection between HARVEY Composer and a HARVEY unit and have transmitted projects to the unit, you cannot make any structural changes to them on the device anymore.

➔ You can make changes to the project structure **in offline mode only**.

The basic method of operation in HARVEY Composer is as follows:

1. First, you create the project offline by placing, wiring, and configuring all function blocks. You can save your parameter settings as presets (or snapshots) for later recall.
2. In the next step, you transmit the project to the HARVEY unit and perform fine tuning of your parameters online.

➔ Note that you can change parameter settings on the HARVEY device only in **online mode**.

In online mode, HARVEY Composer hides the grid to indicate that you currently cannot perform any changes to the project configuration.

➔ When project transmission is complete, the audio will be globally muted for safety reasons:

This is done for security reasons so that a possible configuration error in the project will not damage external devices such as speakers. The sound must be switched on in the HARVEY Composer software via the switch in the status bar.

➔ After transferring a currently active project to a device in the cockpit, the project is in online status (see section "Cockpit" on page 13).

The online / offline status of a project is changed in the status bar of the main window (see section "Main Window" on page 16).

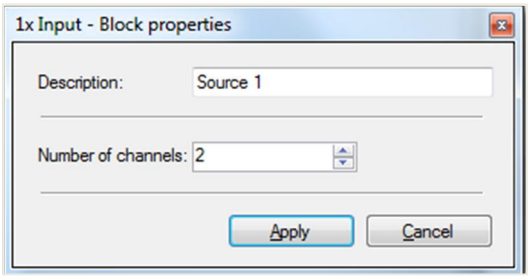
2.8 Creating Projects

To create a new project, first launch the cockpit in HARVEY Composer, then click the **[Create New Worksheet]** button (see "Cockpit" on page 13).

Use drag and drop to place all the required function blocks from the library on the worksheet. If you drop function blocks with inputs or outputs over the worksheet, two windows open automatically and one after the other: The **Block Properties** and the **Connection Editor**.

2.8.1 Block Properties

In the <Block Properties> dialog, you configure a block ID and set the number of channels.



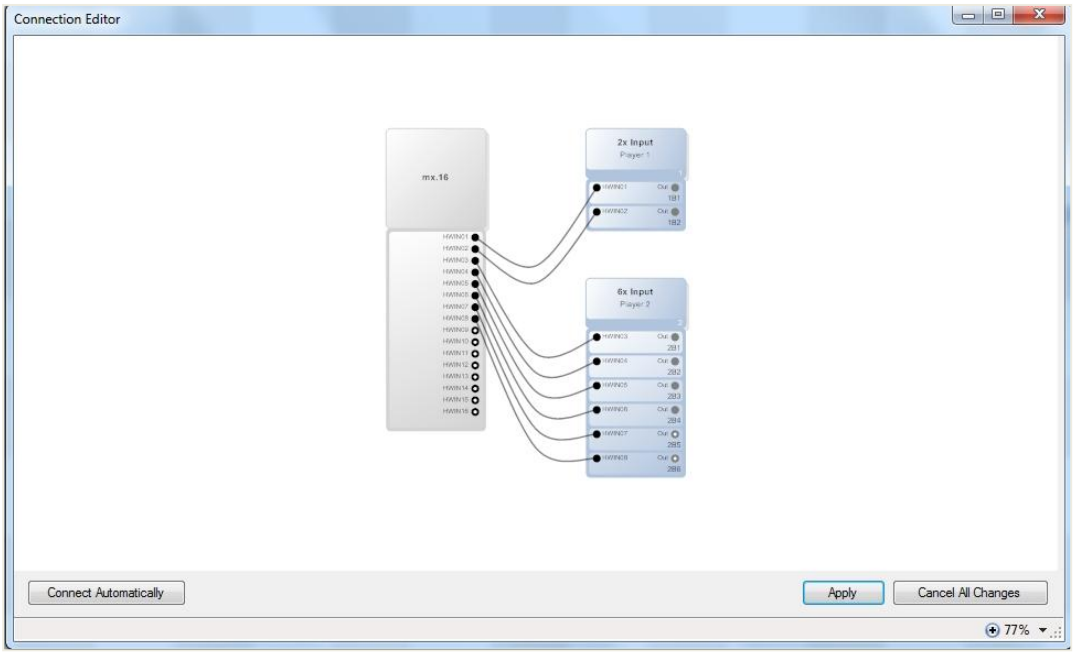
➔ To access the dialog, right-click any function block and select <Properties> from the pop-up menu.

➔ The Block Properties dialog also allows for increasing the channel count at a later time.

➔ To reduce the channel count, you need to manually delete one or more channels in the Channel List (see the “Channel List and Wires” section on page 30).

2.8.2 Connection Editor

In the Connection Editor, you wire the ports of a function block to other function blocks or the ports of the HARVEY device. The editor lists all physically existing inputs and outputs of the system and the block to connect it to.



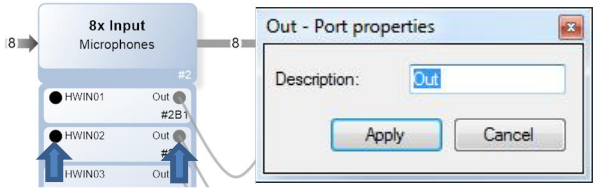
The I/O count depends on the type of function block and the number of available HARVEY device ports. You can click the **[Connect Automatically]** button to have the software auto-wire the ports. In this case, the ports available on the HARVEY device will be wired one by one. You can also perform the wiring manually: Move your mouse pointer over one of the black circles representing the ports on the HARVEY block, then drag to any port of the I/O block. Click the **[Apply]** button to confirm your changes and close the window.

➔ Use the mouse wheel to zoom in and out of the Connection Editor.

If you want to launch the Connection Editor later, double-click the arrow on the left or the right side of a function block (next to the trunk).

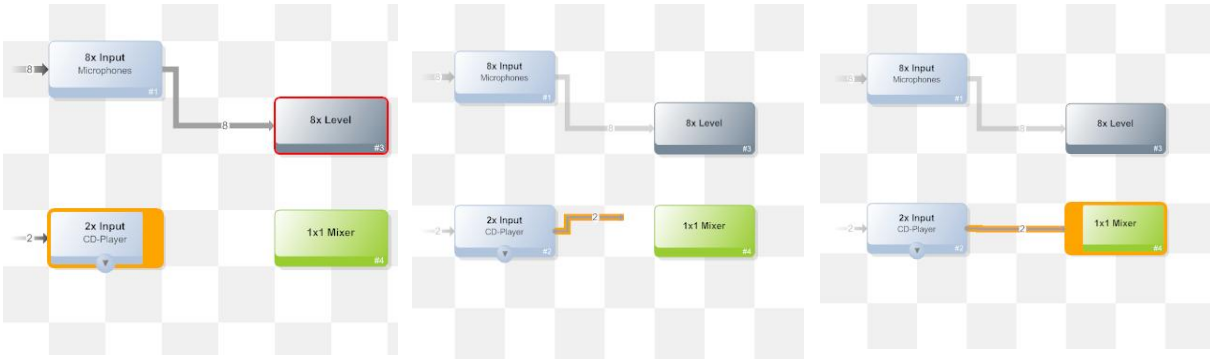
2.8.3 Port Properties

➔ Double-clicking a port opens the **<Port Properties>** dialog where you can enter a new port name.



2.8.4 Wiring Function Blocks

After placing the function blocks on the worksheet, you can wire them to one another.



To do so, move the mouse pointer over the right border of a block until it changes its shape to an arrow. While pressing the left mouse button, drag a line from the source block to the target block, then release the mouse button. The link is marked by a line that adjusts automatically whenever you change the position of an icon or rearrange the icons. The number of links within the drawn trunk is adjusted automatically with regard to the number of channels.

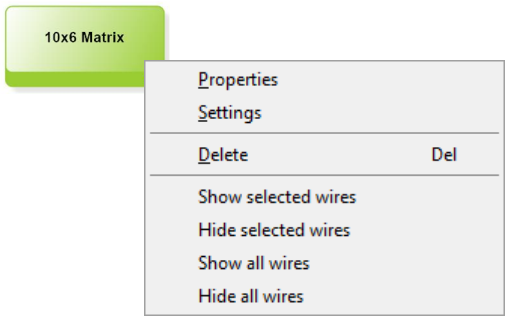
- ➔ A small number on the trunk shows the links contained in it.
- ➔ To remove a trunk, click to select it, then press the <Del> key on your computer keyboard.

You can also insert a new function block right between two existing ones that are interconnected. To do so, drag the new block from the library and drop it onto the trunk between the blocks on the worksheet. The software adjusts the channel count of the new block as appropriate for the trunk. All function blocks that have already been placed on the worksheet will be shifted accordingly (provided the new function block can be placed in the selected position). A preview shows where the adjacent blocks will be moved and if and how audio links will be rearranged.

- ➔ The ("No") symbol ⃠ appears as a mouse pointer if no function block may be stored at this position. Select a different location on the worksheet.
- ➔ There must always be a minimum space of one grid rectangle between any two blocks.
- ➔ At any time, you can rearrange one or more function blocks on the worksheet. To select multiple blocks at a time, either drag a select rectangle around them (lasso selection), or press and hold the <Ctrl> key while clicking each block you want to select.
- ➔ While block inputs accept only a single connection, you can wire one output to multiple inputs at the same time.
- ➔ Individual connections of channels are inserted in the "channel overview" of function blocks (see section “Channel List and Wires" on page 30).

2.8.5 Context Menu of Function Blocks

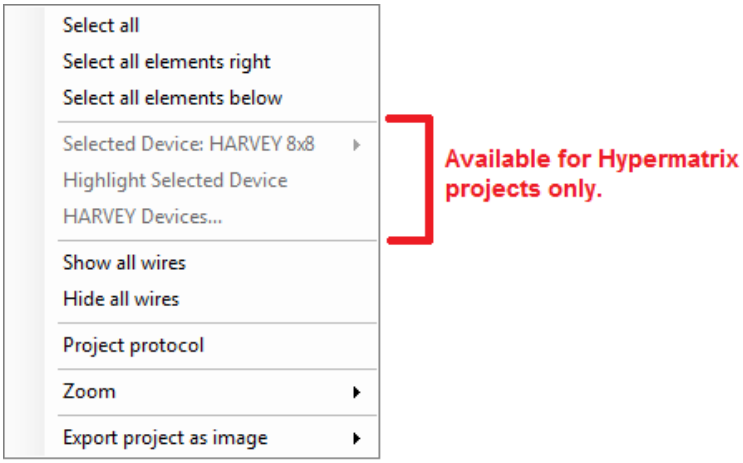
A right-click after pointing at any function block opens a pop-up menu.



- ➔ Under the item "Properties" you get to the block properties. The name, the number of inputs and outputs (and possibly further options) of the block can be determined.
- ➔ The item "Settings" in the context menu leads to the parameters of the block (see page 34).
- ➔ “Delete” leads to a removal of the corresponding function block from the worksheet. Alternatively this can be done by pressing the <Del> key after the selection of a single or a set of blocks.
- ➔ The menu items "Show and hide selected wires" show and hide the channel lists (see page 32) of the selected block and their connected other blocks together with their single wire connections. The menu items "Show and hide all wires" shows the channel lists and single wires globally for the whole worksheet.

2.8.6 Context Menu of Worksheet

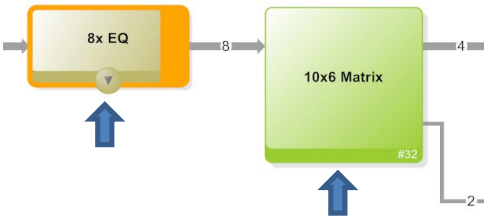
A right-click after pointing into free space on the worksheet opens a pop-up menu with global functions.



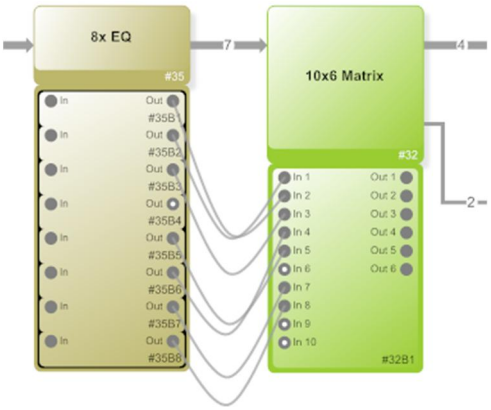
- ➔ The menu items “Select” highlight and prepare a set of blocks to be moved around on the worksheet or to be deleted.
- ➔ The menu items "Show and hide all wires" shows the channel lists (see page 32) and single wires globally for the whole worksheet.
- ➔ The additional functions “Project protocol”, “Zoom” and “Export project as image” are shortcuts which can be found alternatively in the Composer Project menu or the Status bar of the Main Windows (see page 17).
- ➔ There are additional menu items available exclusively for Hypermatrix project types (see page 38).

2.8.7 Channel List and Wires

Click the arrow at the bottom of the block to show or hide the **channel list**. The arrow is displayed only when the mouse pointer is moved over the function block. Similarly, individual wires are visible only when the pointer is moved over a block.



The channel list allows for manually connecting **individual block ports**. Ports and wires are color-coded depending on the category that they belong to. Note that you can wire only ports of the same type or color. Audio lines with audio signals are always displayed in gray, blue lines are level control signals. Green lines carry logic control signals with binary switching states, while brown lines carry data from serial interfaces.



Clicking the **[Audio/Control]** button toggles the worksheet between the two layers. All function blocks that provide no functionality or ports in the selected layer are displayed as transparent; blocks providing functionality or ports in both of the two layers adjust their appearance to show connectivity options in the selected layer.

2.8.8 Deleting Channels

To reduce the channel count of a block, you can delete individual channels in the channel list:

- ➔ Right-click the channel that you want to delete, then select the **<Delete>** item from the pop-up menu. You can also select the channel, and then press the <Del> on your computer keyboard.
- ➔ To select multiple channels, select each channel you need while holding the <Ctrl> key. When you're finished, release the <Ctrl> key. All selected channels will now have a red border. Next, right-click any channel, then select the **<Delete>** item from the pop-up menu or press the <Del> on your computer keyboard.
- ➔ You can also select a contiguous range of channels (for example, channels 9-16). To do so, click the first channel, then press and hold the <Shift> key while clicking the last one.

2.8.9 Groups

With the majority of function-block types, you can group several channels for easier handling. Channels are grouped and ungrouped in the channel list. You configure and operate grouped channels using a channel-group interface. Grouped channels still have their individual input and output ports.

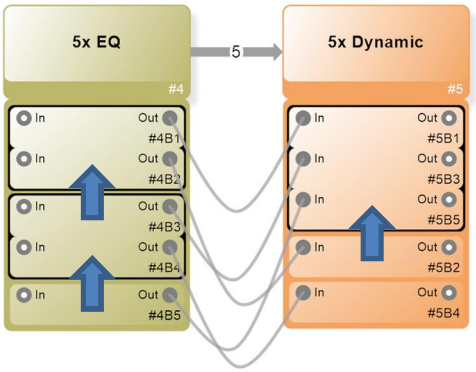
- ➔ The channel list changes to show all channels belonging to a group one after another inside a black frame.
- ➔ Note that grouping will result in rearranging the channels, so channels that have not been neighbors before will now be listed one after another but will keep their original channel ID and wires. When those channels are subsequently removed from the group, they will return to their original position in the list.

To create a new group or break an existing one, first select the channels from the channel list:

- ➔ While holding the <Ctrl> key on your keyboard, click each channel you want to add to or remove from the group. When you're finished, release the <Ctrl> key. All selected channels will now have a red border.
- ➔ Next, to create your group or add selected channels to an existing one, point to one of the selected channels. Right-click the selected channel and select **<Create Group>** from the pop-up menu.
- ➔ If you want to separate an entire group, point to one of the selected channels. Right-click the selected channel and select **<Release a Group>** from the pop-up menu.
- ➔ To remove one or more channels from the group, select the channels to be removed, then right-click and select **<Release From a Group>** from the pop-up menu.
- ➔ You can also select a contiguous range of channels (for example, channels 9-16). To do so, click the first channel, then press and hold the >Shift> key while clicking the last one.

The following example shows two function blocks each with five channels, some of which are grouped:

Channels 1 and 2 and channels 3 and 4 of the EQ block have been grouped; similarly, channels 1, 3, and 5 of the Dynamic block have been grouped. Note that the channels of the Dynamic block have been reordered to provide an integrated view of the resulting group but keep their original channel IDs (#2B1, #2B3, #2B5) as well as their wires to the EQ block.



2.8.10 Settings / Function-Block Parameters

To access the Block Settings dialog, double-click a function block. You can also open this dialog by right-clicking the block and selecting Settings from the pop-up menu.

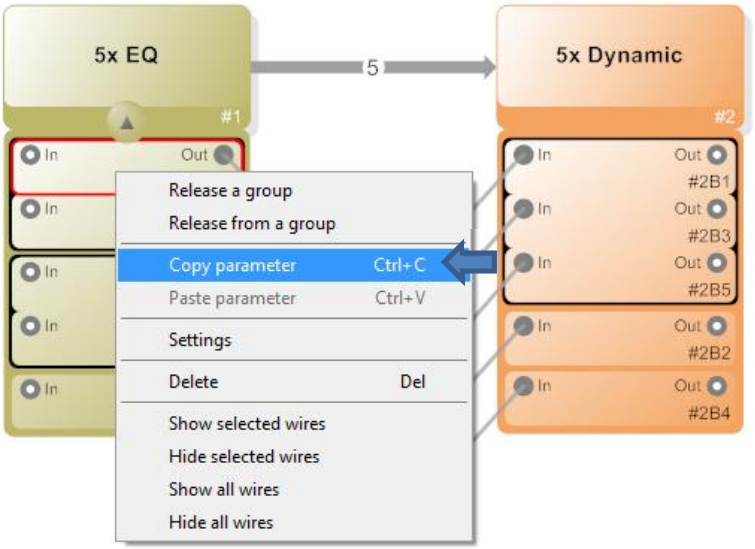
The “Block Settings” provides a separate panel for adjusting each channel or a group of channels. When the project is online, i.e. your computer is connected to a HARVEY device, all changes made here will be immediately applied, so you can hear them at once.

➔ For details on function-block parameters, refer to the “Audio Function Blocks” chapter starting on page 44 and the “Control Function Blocks” starting on page 89, respectively.

2.8.11 Copy & Paste of Parameters

You can copy full parameter sets from a channel or group to another one using the clipboard. This way, you can quickly and conveniently apply equalizer curves or other complex parameter sets:

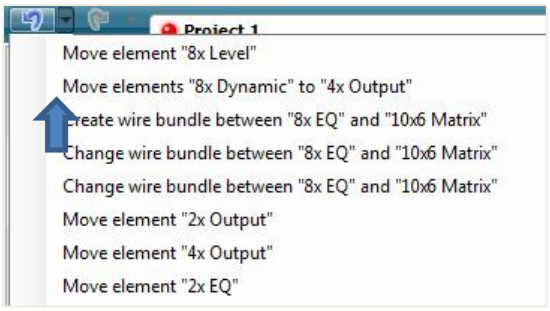
➔ Display the source function block’s channel list and point to the channel the parameters of which you want to copy. Right-click the source channel and select **<Copy Parameter>** from the pop-up menu. Alternatively, you can select the channel and press <Ctrl> + <C> on your computer keyboard.



➔ Point to the channel you want to apply the copied parameter set to. Right-click the target channel and select **<Paste Parameter>** from the pop-up menu. Alternatively, you can select the channel and press <Ctrl> + <V> on your computer keyboard.

2.8.12 Undo and Redo

The **Undo and Redo** lists are history files that store recent edits of a HARVEY Composer project.



- ➔ At the top of the main window, click the blue left arrow once to undo your last action; to redo undone actions, click the right arrow.
- ➔ To view all logged edits, click the triangle next to each arrow. In the list displayed, you can quickly select one specific action to undo or redo all edits stored up to that point in the edit history.
- ➔ Click the <Extras>/<Options>/<Processing> menu item to set the number of edits stored in the undo/redo history.
- ➔ You can set a maximum of 32 edit actions (default: 20).

2.9 Working with Hypermatrix Projects

HARVEY Composer integrates two ways of programming and configuring projects within the HARVEY ecosystem:

1. Single device projects
2. Multi-device projects also known as Hypermatrix projects.

You decide about the project type in the Composer Cockpit (see page 13).

- ➔ Currently there is no way to change the project type after creation.
- ➔ Hypermatrix projects are not limited to but should integrate HARVEY devices with Dante interfaces for full functionality.

There are no basic differences in the workflows of programming single device projects and Hypermatrix projects. Therefore you should be familiar with the basic workflows of HARVEY Composer In addition the following subsections lead you through the programming process of Hypermatrix projects and their specific characteristics.

2.9.1 Overview of Hypermatrix Concept

Hypermatrix projects allow the programming of multi-device projects within a HARVEY ecosystem. All devices part of a Hypermatrix interact via their control Ethernet interface and, if needed, via their Dante interface audio-wise. Thus all Hypermatrix devices present themselves virtually as one large audio and control matrix.

Hypermatrix allows you to accomplish following **use cases**:

- ➔ Distribution of HARVEY devices in different rooms of a building.
- ➔ Scaling of HARVEY interfaces, i.e.for your project you require more interfaces than offered by a single device.
- ➔ Scaling of processing power, i.e. for your project you require more processing power than offered by a single device.
- ➔ A combination of the aforementioned cases.

In contrast to a single project type, the Hypermatrix projects offer following **features**:

- ➔ One HARVEY Composer worksheet for up to ten HARVEY devices.
- ➔ Audio interconnections between audio function blocks are automatically routed via Dante channels without the need to manually configure the routing in Dante Controller. Dante channel endpoints not used by Hypermatrix can be used freely for interconnection with 3rd party Dante devices. These devices are integrated conventionally with the help of Dante Controller.
- ➔ Presets are handled automatically system wide via the standard control network interfaces of the HARVEY devices.
- ➔ Level links are handled automatically system wide via the standard control network interfaces of the HARVEY devices.
- ➔ HARVEY Composer network connections can be established to any HARVEY device of the Hypermatrix.

2.9.2 Prerequisites for a Hypermatrix Installation

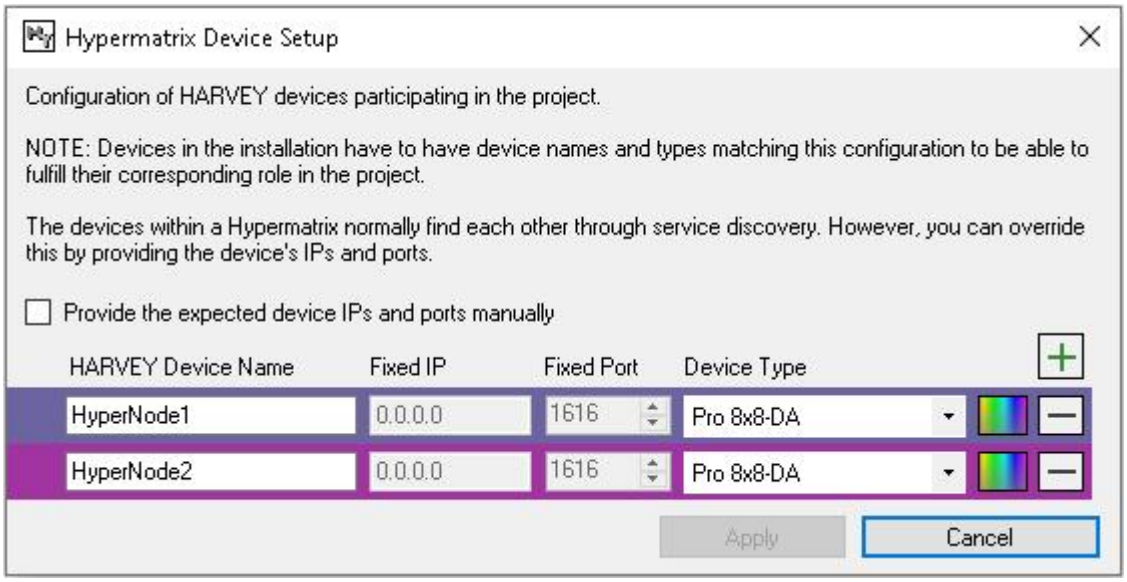
Following requirements have to be fulfilled for a meaningful Hypermatrix installation:

- ➔ A minimum of two HARVEY devices in your installation.
- ➔ Currently the maximum supported number of HARVEY devices is ten; please contact us, if you require more and be sure that we will support you!
- ➔ All HARVEY devices part of a specific Hypermatrix project must be configured for the same control Ethernet network segment.
- ➔ In case you require audio channels to be processed distributed over different Hypermatrix devices those devices require a Dante interface card integrated.

2.9.3 Creating Hypermatrix Projects

Creating a Hypermatrix project follows the same workflow than for a single device project:

- ➔ Invoke the Composer Cockpit (see page 13)
- ➔ Select the project type “HARVEY Hypermatrix”
- ➔ Press the button “Create a new worksheet”
- ➔ The “**Hypermatrix Devices Setup**” window opens upon switching to the worksheet:



This window allows to predefine the number of devices, device names, device types (variants) and a color. Optionally, the IP addresses and ports of the devices can be set.

Number of devices: By default the sub-window opens with the minimum of two device entries. By pressing [+] another Hypermatrix device is appended to the list of devices. By pressing [-] a specific device is deleted from the list.

HARVEY Device Name: In Hypermatrix projects the device name plays an important role. The Hypermatrix device names are stored in the HARVEY Hypermatrix project file. The entry in HARVEY Device Name must match the device name of the physical device (set in Composer Cockpit, see page 13 **[Device Settings]**).

Device Type: This field defines the capabilities of the physical device. The device type of the corresponding device (name) is stored in the HARVEY Hypermatrix project file. It must match the variant of the physical device.

Color: The color visualizes the Composer function blocks' membership of their Hypermatrix device.

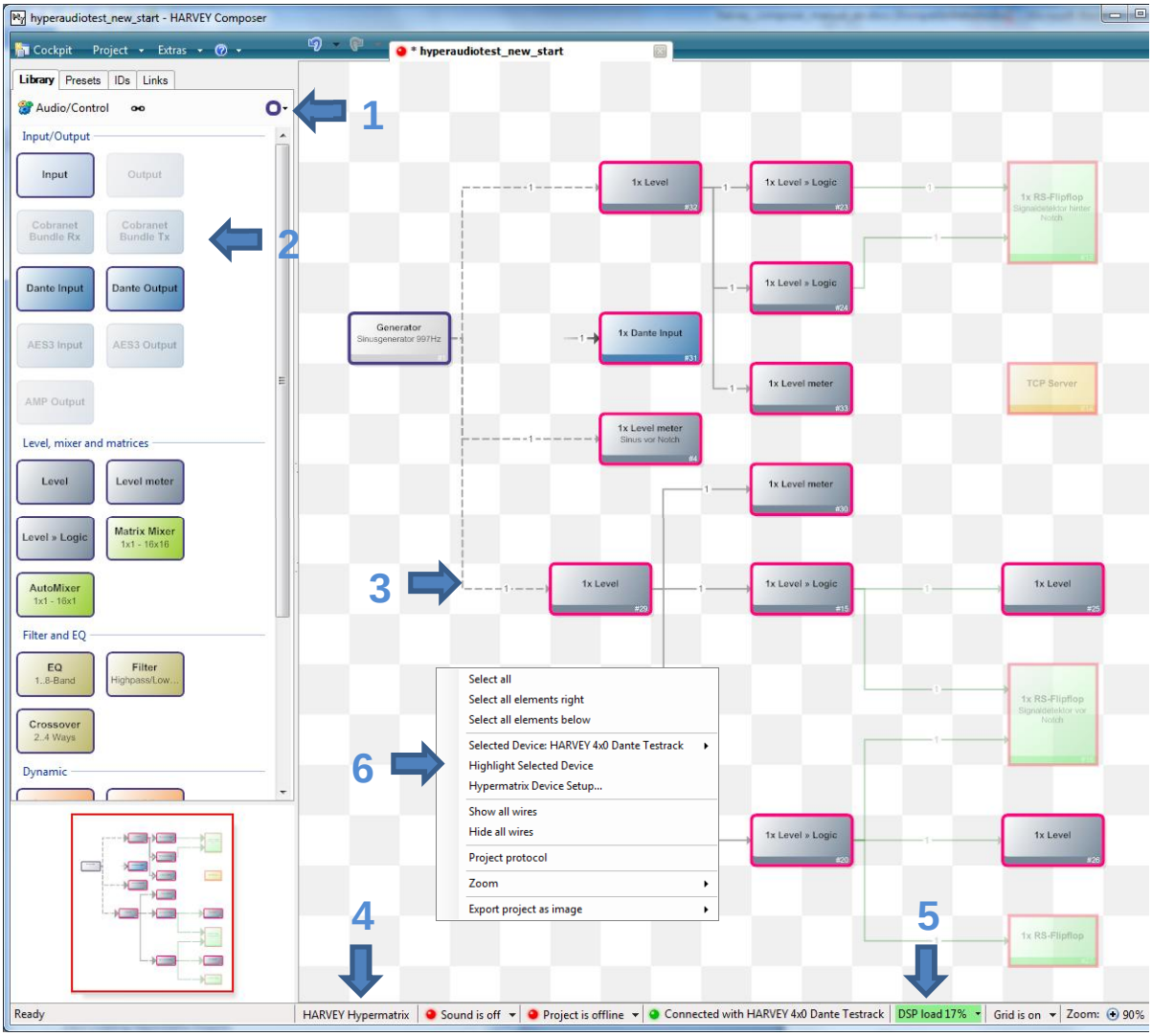
Manual IP and port allocation: This point allows to override automatic device discovery by entering static IP addresses and ports. That may be necessary in some networks which have restrictive routing or switch configurations.

Note: Entering manual addresses here will not change the IP configuration of the devices. It merely informs all hypermatrix participants under which addresses they can expect to find their peers. IP-Address/Port settings entered in the Hypermatrix setup must therefore match those that are configured in the devices. Don't disable automatic device discovery if you have dynamic IP addressing (DHCP) configured in your device settings because the device addresses may then change on power cycle without notice.

➔ Note: The Hypermatrix Device Setup can be reopened from the menu bar (Project/Hypermatrix Device Setup) and from the status bar (HARVEY Hypermatrix).

2.9.4 Programming and Configuration Hypermatrix Projects

The programming and configuration of a Hypermatrix project basically is similar to that of a single device project. Nevertheless, in detail there are some differences which are explained in this subsection.



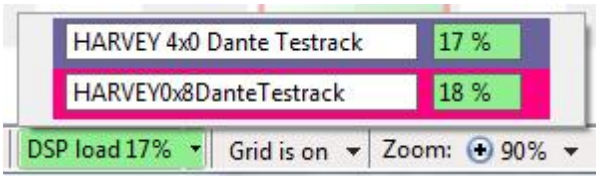
Device selection (1): On the right top of the sidebar´s library tab there is a colored circle which after selection drops down the Hypermatrix devices you have set up during project creation (see page 38, Hypermatrix Device Setup). Here you choose, for which device you currently want to place function blocks onto the worksheet.

Library (2): The function blocks in the library have a frame outline with the same color as the currently chosen device – only those blocks fitting to the capability of the device type are marked as active.

Wires (3): When connecting function blocks, which are placed for different devices on the worksheet (by changing the device selection and drag-dropping from the library), inter-device wires are shown as a dashed line. All dashed lines represent network links – audio connections are routed via the Dante network and control wires are routed via the control network.

Statusbar [HARVEY Hypermatrix] (4): When pressing that button, you re-invoke the Hypermatrix Device Setup (see page 38). That enables you to subsequently change your definition of Hypermatrix devices.

Statusbar [DSP load] (5): The DSP load is shown for the currently chosen device. That means, if you change the current device selection (1), than the load figure updates its value. When clicking on the [DSP load] button a list of all Hypermatrix devices and their load figures is displayed.



Worksheet context menu (6): When right-clicking into a free area of the worksheet the global worksheet context menu opens (also see page 31), which offers, in contrast to single device projects, three more Hypermatrix specific menu entries:

<Selected Device> offers the same function as the device selection in the sidebar´s library tab (1).
<Highlight Selected Device> highlights all blocks on the worksheet assigned to the currently selected device for a better overview in projects.
<Hypermatrix Device Setup> is a further shortcut to re-invoke the Hypermatrix setup (see page 38).

2.9.5 Transferring Hypermatrix Projects from and to HARVEY Devices

Before transferring the Hypermatrix project to a system of HARVEY devices, the following two major things must be considered:

- ➔ All devices configured in the Hypermatrix Device Setup (see page 38) must be physically and mutually accessible via the control Ethernet network from HARVEY Composer.
- ➔ As a prerequisite of transferring a Hypermatrix Composer project to the physical Hypermatrix system, each physical device must be of the device type and must have their device name set according to the Hypermatrix Device Setup settings in the Hypermatrix project.
- ➔ See page 42 for further troubleshooting tips.

Transferring a Hypermatrix project follows the same workflow like transferring a single device project to the system:

- ➔ Open the Hypermatrix project which you designed offline.
- ➔ Open the Composer cockpit, choose one of the devices which should be a member of the Hypermatrix project and connect it. It does not matter which device you choose – you merely have to care about the fact that it is a device which has same name and type as defined in the Hypermatrix Device Setup.
- ➔ Press button **<Transfer active project to the device>**.
- ➔ Remember, like valid for single device projects that the sound is switched off globally after a project transfer. Switch it on manually in the status bar of the main window.

When a transfer of a Hypermatrix system to the HARVEY Composer is intended:

- ➔ Open the Composer cockpit, choose one of the devices which is a member of the Hypermatrix project and connect it. Again, it does not matter which device you choose.
- ➔ Press either the button **<Transfer project of the system in a new working sheet>** or **<Save backup from project of HARVEY on PC>**.

2.9.6 Presets in Hypermatrix Projects

Presets in hypermatrix projects are basically defined in the same way as in one-device projects. All block parameters of the project are available for the definition of a preset. Thus, Presets act in the hypermatrix across all devices.

Synchronizing the Preset Execution

A preset call is originally always triggered on a single Hypermatrix device and distributed from there to all other devices in the Hypermatrix. The preset call can be done via the usual mechanisms:

- ➔ HARVEY Composer,
- ➔ a preset block in the project,
- ➔ an H-Text command,
- ➔ the web interface,
- ➔ as Start Preset during system startup
  (Attention: See below *Partial Preset Execution*)

After distribution in the network, a preset is executed individually on all hypermatrix devices specifically for the blocks assigned to each device.

Partial preset execution

The following scenarios can result in a preset not being executed on all Hypermatrix devices:

- ➔ The Hypermatrix device on which the preset was triggered cannot reach all other devices in the network (e.g. because they are switched off or the network connection is disconnected).
- ➔ The preset is executed as a Start Preset. In this case, the preset is not synchronized on all devices, but only individually on the starting device with the block parameters assigned to it.

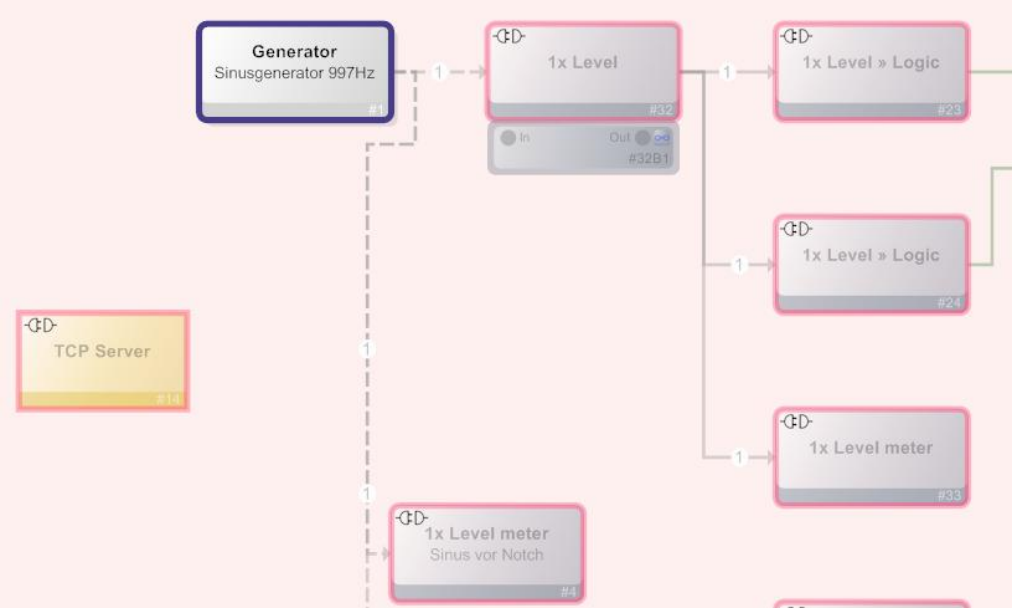
  A cross-device execution of the Start Preset therefore only takes place if *all* devices in the hypermatrix restart.

In the described cases, this can lead to a preset execution that is not completely synchronized and, as a result, to unexpected constellations in the project configuration.

2.9.7 Troubleshooting

Hypermatrix devices not mutually accessible in network

In a situation where physical member devices of a Hypermatrix are not mutually accessible in the control network you'll see a "disconnect" icon ( ) in the left top border of all function blocks of an online project:



This example situation shows an accessible blue device and a not accessible red device.

In that situation check following points:

- o Are all Hypermatrix devices powered and switched-on? Check green LED signaling on their front panel.
- o Are all network cables attached to all devices and the network infrastructure? Check link and activity signaling of the control RJ45 ports.
- o Are all devices configured for the same IP segment? Check the network settings with the help of the front user interface of the devices.

- Do all devices have the same firmware version installed? Check with the help of the front user interface of the devices.
- Are all devices loaded with the same project file? Check with HARVEY Composer by transferring each project from each device and compare.
- Do all devices have their name set identical to the name defined in the Hypermatrix project? Check with the help of the front user interface of the devices and compare to your Hypermatrix project setting.
- Are all devices of a hardware variant identical to the type defined in the Hypermatrix project? Check with the help of the front user interface of the devices and compare to your Hypermatrix project setting.

Dante interconnection between devices does not work – no audio

For Dante problems check following points:

- Are all network cables attached to the devices' Dante RJ45 ports and the network infrastructure? Check link and activity signaling of the Dante RJ45 ports.
- .Are all Hypermatrix devices foreseen as Dante devices visible within Dante Controller?
- Are all Dante devices configured for the same IP segment? Check within Dante controller on the device's status tab.
- Are all Dante routes set correctly by HARVEY? Check within Dante controller.

3 Audio Function Blocks

Audio function blocks handle audio in the HARVEY DSP environment.

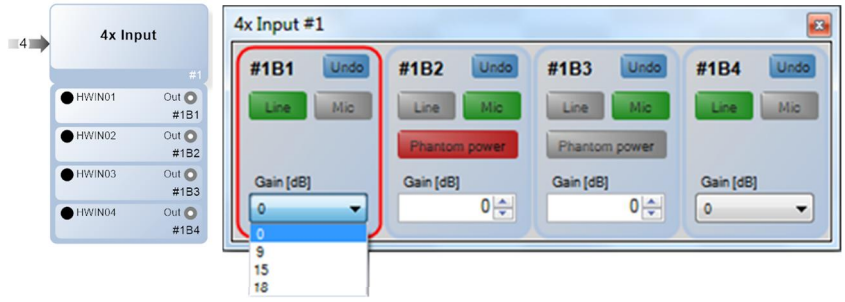
➔ Some audio function blocks provide control layer connections.
Control ports are listed below the description per block in a table.

3.1 Audio Function Blocks: Inputs and Outputs

This section describes all audio function blocks that provide input or output ports.

➔ Some function blocks will be available on the HARVEY device only when the “CobraNet” or “Dante” interfaces are integrated.

3.1.1 Input (Audio Function Block: Input / Output)



The input function block transfers the hardware analog audio inputs of the HARVEY device to the software DSP level and provides all input amplifier parameters. This function block can contain several input channels at the same time (depending on the device variant), whereby each of these software audio inputs must be assigned an exclusive HARVEY audio input (hardware port). A multiple assignment is inadmissible, each of the hardware ports can only be assigned once. This allows a maximum of as many software audio inputs in a project as the selected device variant has audio inputs. It does not matter whether, for example, 16 function blocks "1x input" or 4 function blocks "4x input" are used, or any other combinations. Multiple channels can be managed in groups.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Use the **[Line]** or **[Mic]** buttons for setting the sensitivity of the corresponding hardware input to line or microphone level. The selected operating mode is known by the green button.

In **Mic mode**, the input has a switchable [phantom power] (P48). Active phantom power is displayed in red.

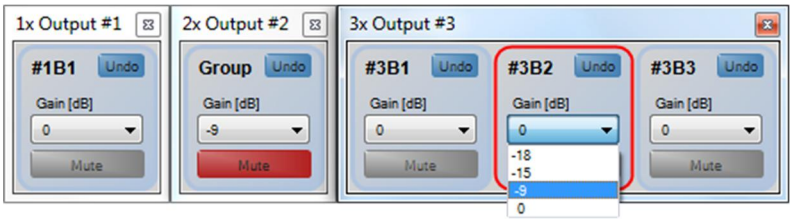
➔ The other parameters that can be set differ depending on the device variant and the project type set (see Setting Project Type in the Cockpit on page 13 and Main Window on page 16 and the HARVEY Device Variants on page 139):

Property	**HARVEY Pro**	**HARVEY mx.16[Dante][Cobranet]**
Number of inputs	HARVEY 16x8: 16 HARVEY 8x16: 16 All ports Mic-capable	16 Only ports 1..8 Mic-capable
<Gain [dB]> for Line-mode	0 dB, + 6 dB, +9 dB, +15 dB	0 dB, +9 dB, +15 dB, +18 dB
<Gain [dB] for Mic-mode	0..60 dB in 3 dB steps	0, 10..65 dB in 1 dB steps

➔ Identical absolute gain values of directly connected inputs (in line mode) and outputs represent a "unity gain" setting (eg +6 dB in the input and 6 dB in the output).

➔ The block "Input" does not provide any additional connections in the control layer.

3.1.2 Output (Audio Function Block: Input / Output)



The function block “Output“ transmits audio signals from the software DSP level to the HARVEY hardware analog audio outputs and provides all output amplifier parameters. This block can contain several output channels at the same time (number depending on the device variant), whereby each of these software audio outputs must be assigned an exclusive HARVEY audio output (hardware port). A multiple assignment is inadmissible, each of the hardware ports can only be assigned once. This allows a maximum of as many software audio outputs in a project as the selected device variant has audio outputs. It does not matter whether, for example, 16 function blocks "1x output" or 4 function blocks "4x output" are used or any other combinations. Multiple channels can be managed in groups.

➔ The HARVEY Composer software automatically forms a group when the "Output" function block with multiple software outputs is created in the project. The grouped outputs are then controlled together in the block settings.
The group can be resolved in the channel overview to make different settings (see section "Groups" on page 33).

Clicking the **[Undo]** button will undo all changes made since the settings window was opened.

Use the <Gain [dB]> drop-down menu to set the maximum output level of audio outputs. When set to 0 dB, the maximum output level of the device is set.

[Mute] mutes an audio output. Active mute is displayed in red.

➔ The other parameters that can be set differ depending on the device variant and the project type set (see setting Project Type in the Cockpit on page 13 and Main Window on page 16 and the HARVEY Device Variants on page 139):

Property	**HARVEY Pro**	**HARVEY mx.16[Dante][Cobranet]**
Number of Outputs	HARVEY 8x16: 16 HARVEY 16x8: 8	16
<Gain [dB]>	0 dB, -6 dB, -9 dB, -15 dB	0 dB, -9 dB, -15 dB, -18 dB

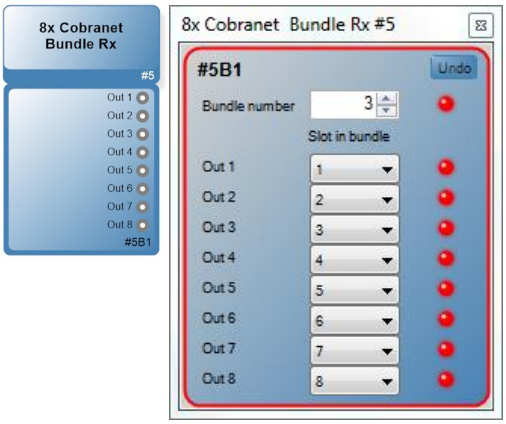
➔ The negative sign of the gain values means that the value is to be seen as attenuation. In the device hardware, however, is generally amplified, so that the largest output level and gain in the position "0 dB" is achieved.

➔ Identical absolute gain values of directly connected inputs (in line mode) and outputs represent a "unity gain" setting (eg +6 dB in the input and 6 dB in the output).

➔ The function block "Output" provides additional connections in the control layer:

Parameter	Control Connection	Range
Mute	Logik Input	On/Off

3.1.3 CobraNet Rx (Audio Function Block: Input / Output)



The “CobraNet Rx” function block converts audio incoming at the CobraNet interface to the software DSP domain of the HARVEY mx.16 device type. Each CobraNet Rx block receives a full CobraNet bundle comprising 8 audio channels (or slots) and makes these audio channels available at its output ports for further processing. You can freely route the slots of a CobraNet bundle to the output ports of the function block.

- ➔ This feature is available only for HARVEY mx.16 units implementing the CobraNet interface.
- ➔ Note that you can use a maximum of 8 CobraNet Rx blocks in each project.
- ➔ Be sure not to use CobraNet and other Ethernet network protocols in the same environment!

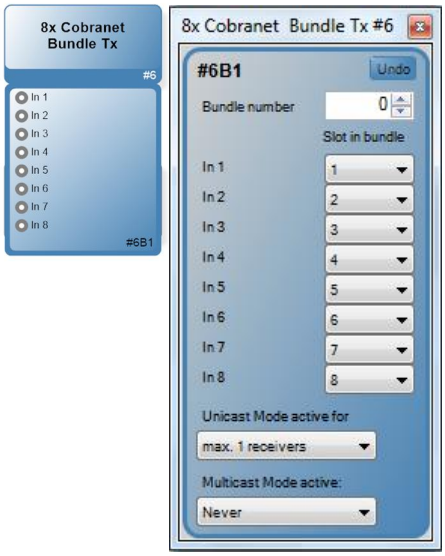
Clicking the **Undo** button will undo all changes made since the Settings window was opened.

Using the **<Bundle Number>** field, you can select the CobraNet bundle you want to use. During audio transmission, the indicator on the right of the input box will be green; if there are no incoming data, the indicator will be red.

You can assign each of the output ports (Out 1-8) of the function block with a CobraNet slot using the drop-downs in the **<Slot in Bundle>** column. Note that you cannot link a slot to multiple output ports. The indicator next to each output will light green at successful reception or red when there are no incoming data.

- ➔ The CobraNet Rx block offers no additional control ports.

3.1.4 CobraNet Tx (Audio Function Block: I/O)



The “CobraNet Tx” function block routes audio from the DSP domain of the HARVEY mx.16 to a CobraNet network. A CobraNet Tx block provides 8 port inputs and outputs the audio as a full CobraNet bundle containing 8 slots (audio channels) at the CobraNet interface. You can freely route the 8 input ports to the 8 slots made available by the function block.

- ➔ This feature is available only for HARVEY mx.16 units implementing the CobraNet interface.
- ➔ Note that you can use a maximum of 4 CobraNet Tx blocks in each project.
- ➔ Be sure not to use CobraNet and other Ethernet network protocols in the same environment!

Clicking the [**Undo]** button will undo all changes made since the Settings window was opened.

Using the **<Bundle Number>** field, you can select a CobraNet bundle as a target for signal routing.

You can assign each of the input ports (In 1-8) of the function block with a CobraNet slot using the drop-downs in the **<Slot in Bundle>** column. Note that you cannot link a slot to multiple input ports.

In the **<Unicast Mode Active For>** drop-down menu, you can set the maximum number of CobraNet receivers (1-4) triggering bundle transmission in unicast mode.

In the **<Multicast Mode Active>** drop-down menu, you set the conditions that trigger bundle transmission to the CobraNet network in multicast mode (Never, From 1-4 Receivers, Always).

- ➔ The CobraNet Tx block offers no additional control ports.

3.1.5 Dante Input/Output (Audio Function Blocks: Input / Output)

The Dante Input and Dante Output Function Blocks provide the HARVEY software with port inputs and outputs for audio signal exchange with the Dante interface. The function blocks may contain multiple input and output channels simultaneously, with each of the software ports having to be assigned an exclusive Dante port. A multiple assignment is illegal, each of the Dante input or output ports can only be assigned once. As a result, the maximum number of Dante inputs and Dante outputs is possible in a project that are permitted by the respective project type

- ➔ Only available with installed Dante interface in the HARVEY Pro and HARVEY mx.16.
- ➔ A Dante network can be mixed with other Ethernet network protocols.
- ➔ The number of available Dante input and output channels depends on the selected project type or device variant (see setting Project Type in the Cockpit on page 13 and Main Window on page 16 and the HARVEY Device Variants on page 139):

Property	HARVEY Pro Dante	HARVEY mx.16 Dante
Number of Dante Inputs / Outputs	64 Input channels 64 Output channels	16 Input channels 16 Output channels

- ➔ The function blocks "Dante Input" and "Dante Output" do not provide function block settings in the HARVEY Composer software.
  All settings are made using the Dante controller software from Audinate.
- ➔ The function blocks do not provide any additional connections in the control plane.

3.1.6 AES3 Input/Output (Audio Function Blocks: Input / Output)

The AES3 Input and AES3 Output Function Blocks provide the HARVEY software with port inputs and outputs for audio signal exchange with the AES3 interface. The function blocks may contain multiple input and output channels simultaneously, with each of the software ports having to be assigned an exclusive AES3 port. A multiple assignment is illegal, each of the AES3 input or output ports can only be assigned once. As a result, the maximum number of AES3 inputs and AES3 outputs is possible in a project that are permitted by the respective project type

- ➔ Only available with installed AES3 interface in the HARVEY Pro device
- ➔ The number of available AES3 input and output channels is limited to 2.
- ➔ The function blocks "AES3 Input" and "AES3 Output" do not provide function block settings in the HARVEY Composer software.
- ➔ The function blocks do not provide any additional connections in the control plane.

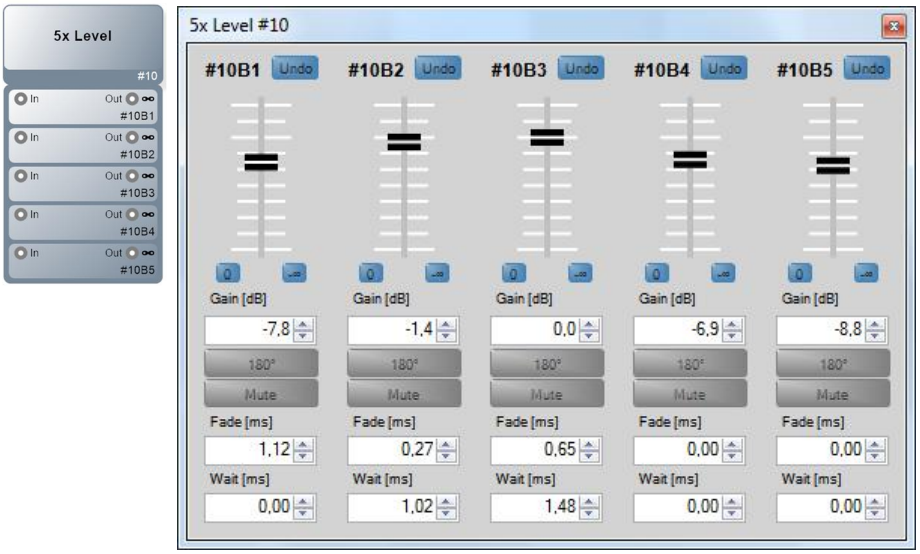
3.2 Audio Function Blocks: Level, Mixer, and Matrices

This section describes components such as faders, meters, the logic mixer/matrix, and the auto mixer.

These components assist in routing and monitoring signals and allow for using logic operations and mixing audio.

➔ If there are not enough inputs or outputs, HARVEY Composer will notify you accordingly.

3.2.1 Level (Audio Function Block: Level, Mixer, and Matrices)



The function block “Level” provides for gain adjustment, phase inversion, and muting of up to 64 audio channels. Each channel has its dedicated port output, meaning that there is no summing function.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

➔ For summing channels, use a downstream Mixer/Matrix block.

➔ This function block allows for grouping channels and also supports the link function (see page 23).

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **channel fader**, you can set the signal level from –99.9 dB to +10 dB in steps of 0.1 dB. The bottom-stop setting (–∞) is equivalent to muting the channel. Using the blue **0** and **–∞** buttons allows for configuring a level of 0.0 dB or –∞, respectively. You can view the current gain setting in the **<Gain [dB]>** field. To change the setting, click into the field and enter the desired value using your keyboard, or use the tiny arrow buttons.

Clicking the **[180°]** button inverts the polarity of the audio. While the function is enabled, the button will be green.

Clicking the **[Mute]** button will mute the port output. While mute is on, the button will be red.

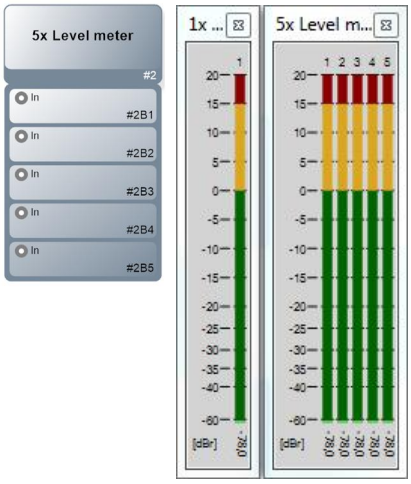
Using the **Fade [ms]** control, you can set the level-fadeover time when changing presets. **Wait [ms]** determines the time to wait before the fadeover starts. You can set times between 0.00 and 15,000.00 ms in 0.01-ms steps.

➔ The Level block offers the following additional control ports:

Parameter	Control Port	Value Range	Status (1)
Mute	Logical Input	On/Off	Current Mute state
Level	Level Input	Preset Range	Current level gain value

(1) The logic inputs return the current parameter state as status to the attached block (e.g. RC4 can handle status to display current level value on LED ring).

3.2.2 Level Meter (Audio Function Block: Level, Mixer, and Matrices)

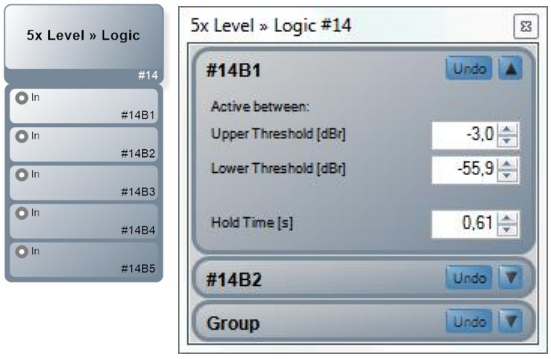


The Level Meter function block provides for visual signal monitoring. The function block measures the incoming audio and shows the level on a dBr scale (relative project-reference level) in the Settings window of the block. The block has no port outputs.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

- ➔ There are no configurable parameters.
- ➔ The <dBr> reference level configured in Extras / Options automatically affects the meters of all Level Meter function blocks.
- ➔ The Level Meter block offers no additional control ports.

3.2.3 Level » Logic (Audio Function Block: Level, Mixer, and Matrices)



The “Level » Logic” function block evaluates incoming audio and sets the logic output to 1/TRUE or 0/FALSE depending on the conditions defined for the input level. This function block provides a single port input on the audio layer and a single logic-port output on the control layer. The block is capable of handling multiple channels at the same time.

The number of input ports automatically adjusts as required. However, you can also set it manually from the Settings window.

➔ The project-reference level configured in Extras / Options automatically affects the threshold settings in this function block. (Refer to page 16)

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

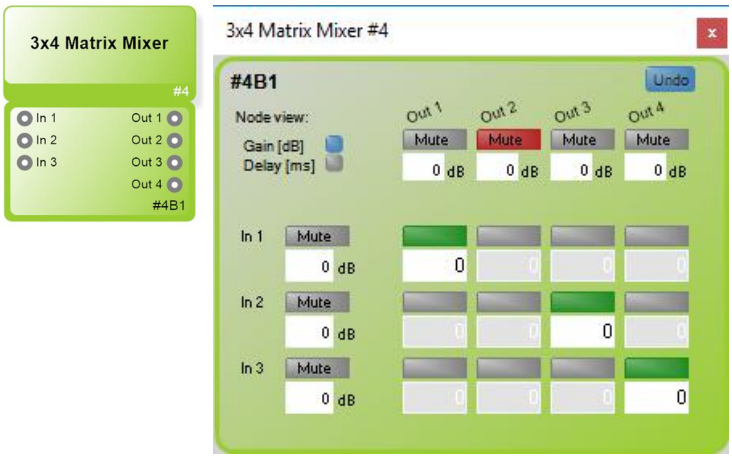
By configuring the **Upper Threshold [dBr]** and **Lower Threshold [dBr]** settings, you can define a level range, which the audio input has to fall into in order to set the logical output to 1/TRUE. Levels falling outside this range set the logical output to 0/FALSE. You can set any range between +∞ (maximum system level) and –∞ (no signal present). The numerical threshold levels of the range vary depending on the configured reference level.

The **Hold Time [s]** parameter allows for configuring a hold time that will be applied as soon as the incoming audio falls outside the configured range. You can set hold times between 0.25 and 10 seconds.

➔ The Level » Logic block offers the following additional control ports:

Parameter	Control Port	Value Range
Metered Audio Level (Condition)	Logical Output	On/Off

3.2.4 Mixer/Matrix (Audio Function Block: Level, Mixer, and Matrices)



The “Matrix Mixer” block provides two functions within the same block: It mixes incoming audio signals and makes the new output signal available at one of its output ports and also routes port inputs to any port outputs using a routing matrix. This way, when used in combination with an upstream Level block, you can create a simple downmix or make different mixes available at the various port outputs. The block can also be used as a simple patch bay. When implementing various system scenarios using presets, this function block becomes an advanced control center thanks to its flexibility.

You can configure any combination of 1-80 inputs and outputs with the maximum number of matrix nodes being limited to 2304. That limit would be reached by a 48x48 matrix, but asymmetric configurations with higher input or output count are still possible, such as 80x28.

The number of input and output ports automatically adjusts as required by wire bundles being dragged on the matrix. However, you can also set it manually from the Settings window.

➔ This function block offers ***3 block options*** through its Settings window. This way, all nodes allow for configuring adjustable gain or gain plus delay settings:

1: Simple mix matrix

2: Mix matrix with gain

3: Mix matrix with gain and delay

➔ To select or remove input channels or copy their parameters on the channel list, click the circular input-port icon.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Clicking the **[Mute]** button will mute the input or output port. While mute is on, the button will be red.

Using the input fields located below the Mute button, you can set the **level** between –99.9 dB and +10 dB in steps of 0.1 dB for each **input and output**. The –∞ setting (below –99.9 dB) is equivalent to muting the channel.

The node buttons allow for **connecting** input ports and output ports. Enabled connections are shown as green buttons.

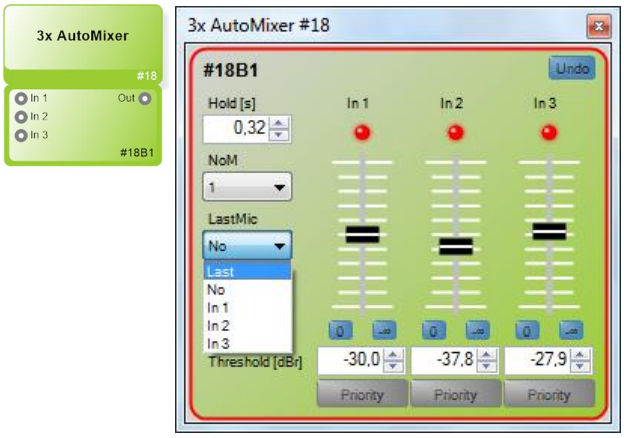
The **Mix Matrix with Gain** provides additional gain-input fields below the node buttons (–∞, –99.9 dB…+10 dB gain in 0.1-dB steps).

With the „**Mix Matrix with Gain and Delay**, you can additionally set signal delays for each connection from 0 to 1000 ms in 0.02-ms steps.

Also, in the Mix Matrix with Gain and Delay, using the **<Gain [dB]>** and **<Delay [ms]>** options in the **<Node View>**, you can set which of the two parameters you can view and adjust in the input fields below the nodes.

➔ The Mixer/Matrix block offers no additional control ports.

3.2.5 AutoMixer (Audio Function Block: Level, Mixer, and Matrices)



The "AutoMixer" is a function block specifically designed for microphone conferences. It is used for automatically downmixing multiple input signals to a single port output depending on the input levels. Each input channel includes an adjustable gate. The function block sums the signals from all open channels and sets the master output level automatically with regard to the channel count in order to prevent feedback. You can adjust the maximum number of channels that can be open at the same time. In addition, you can configure a priority channel; when that channel is open, all other channels are muted. Similarly, you can set one channel to remain open during pauses. The AutoMixer block can be configured with 1-16 channels.

The number of input ports automatically adjusts as required. However, you can also set it manually from the Settings window.

- ➔ The project-reference level configured in Extras > Options automatically affects the threshold settings in this function block. (Refer to page 16)
- ➔ To select or remove input channels or copy their parameters on the channel list, click the circular input-port icon.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **channel fader**, you can set each channel's **gate threshold** in steps of 0.1 dB. You can also view and change the threshold in the input fields below the faders. This setting is relative to the configured dBr reference level of the project. Selecting –∞ disables the gates, meaning that the channel stays open at all times. Using the blue **[0]** and **[–∞]** buttons allows for configuring a level of 0 dBr or –∞, respectively. The **indicator** located above each fader is red when the gate is closed or green when the channel is open.

The hold time set in **<Hold [s]>** determines how long the gate will remain open when the level has fallen below the threshold. It can be set between 0.01 and 10 seconds.

The **<NoM>** ("Number of Open Microphones") drop-down menu sets the maximum number of channels that can be open at the same time. When enabling this function, the selection is based on the switching order (FIFO).

The **<Last Mic>** drop-down menu allows for setting a channel that will remain open even when all gates are closed.

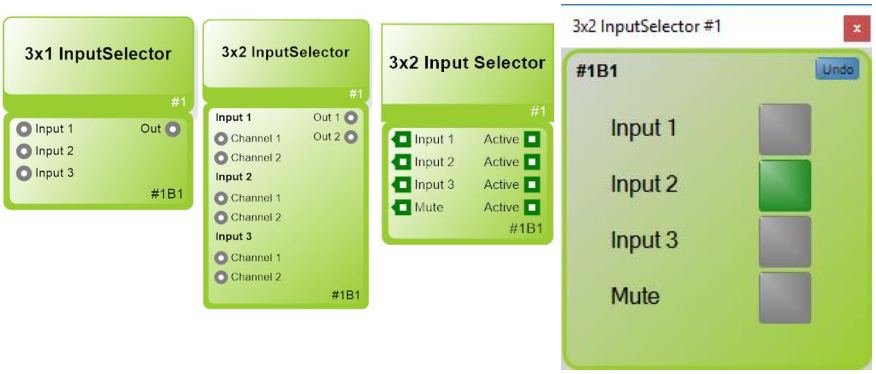
Using the **[Priority]** button, you can set the priority channel (for example, the moderator's channel).

When the priority channel is open, all other channels will be muted.

➔ The AutoMixer block offers the following additional control ports:

Parameter	Control Port	Value Range
Active (available per channel)	Logical Output	On/Off

3.2.6 Input Selector (Audio Function Block: Level, Mixer, and Matrices)

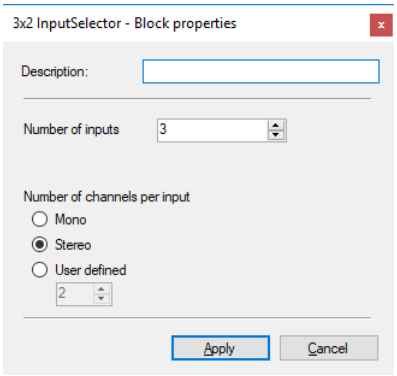


The function block “Input Selector” allows the user to select of one out of multiple audio inputs. The selected input is routed to the block output. One or more audio channels can be configured for each input, which switch simultaneously when an input is selected. The transition between two inputs uses crossfading, to avoid any crackling that may occur when switching between different audio signals.

In addition to the configured inputs, there is the choice “Mute” in all control interfaces of the block. “Mute” clears any previous selection and the output of the block is then muted.

In the control view of the worksheet, logic inputs appear for each selection choice, which can trigger the input selection. The currently active selection is signalled via the corresponding “Active” output.

Block Properties

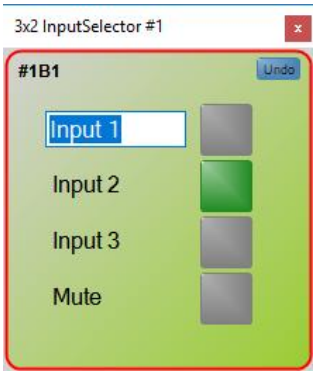


In **Description** a name can optionally be assigned to the block. This name also appears Web Interface when it is generated from the project.

Number of inputs allows the definition of the selectable input count of the block.

Under **Number of channels per input** you can define how many channels are assigned to one input. The popular variants Mono and Stereo can be selected directly for simplification.

Block Settings



The active input can be selected using the radio buttons provided in the settings dialog or the selection can be deleted via **[Mute]**.

[Undo] resets the input selection to the state in which the dialog was opened.

The inputs and the mute choice can be renamed by clicking on the identifier. The input naming corresponds to the control port names in the worksheet and the button labels in the web interface, if the block has been added there.

Controlling of the input selection outside of HARVEY Composer is available through the **Web Interface** and the **H-Text** protocol over serial ports or network.

➔ The „Input Selector“ offers the following additional control ports:

	Name	Control Port	Value Range	Function	Status (1)
Per Input #	Quelle #	Logical Input	On/Off	Select Input #	Is Input # selected?
	Active	Logical Output	On/Off	Signals if Input # is selected	-/-
	Mute	Logical Input	On/Off	Unselect all Inputs	Is Mute aktive?
	Active	Logical Output	On/Off	Signals if Mute is selected	-/-

(1) The logic inputs return the current parameter state as status to the attached block (e.g. to trigger color switching on an attached RC4 or RC12).

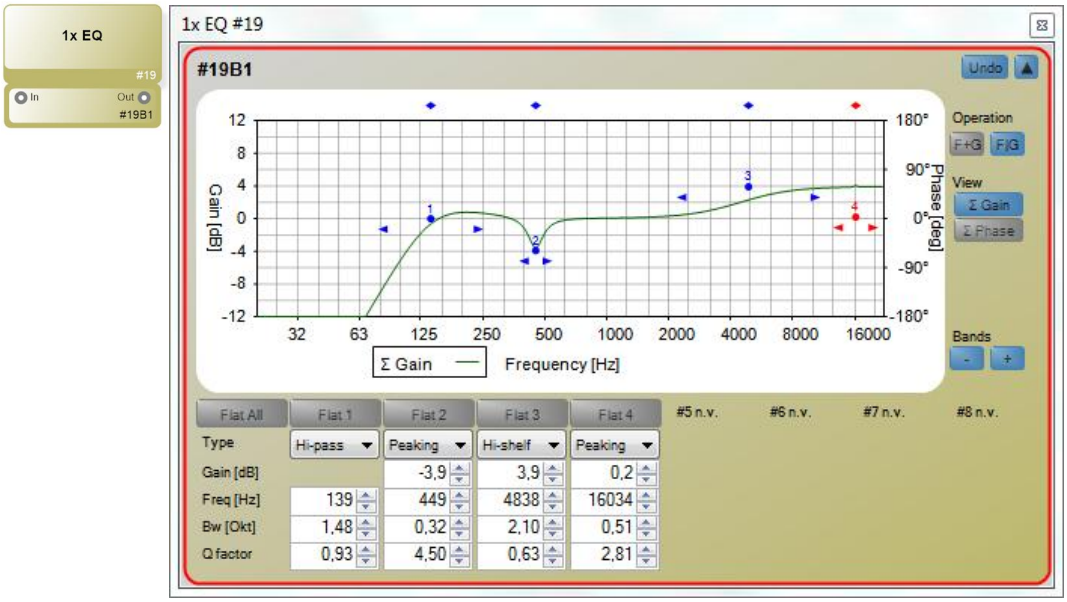
3.3 Audio Function Blocks: Filter and EQ

This section describes all function blocks used for frequency or phase processing.

Depending on the filter type and setting, channel filters may produce frequency-depending phase distortion. (Frequency dependence results from the use of recursive filters.) This results in frequency-depending latencies typically in the range of the respective cutoff frequencies.

➔ Additional latency caused by filters is not automatically compensated; therefore, you may need to perform manual compensation using all-pass filters.

3.3.1 1..8-Band EQ (Audio Function Block: Filter and EQ)



The “EQ 1..8” function block offers up to 8 independent fully parametric filters per channel. Each block includes up to 64 channels and also handles channel groups.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

The individual **bands** are adjusted either in the diagram using the mouse or by entering the corresponding values in the lower part of the window. Two graphical operating modes are available in the right-hand side of the window: In the **[F+G] Mode**, you can configure the frequency and gain settings by moving the EQ-band dot horizontally and/or vertically; In the **[F|G] Mode**, you can move the dot only vertically for adjusting the gain while the frequency setting is made using a diamond on top of the diagram. The bandwidth setting is visualized by two arrows located on each side of the dot. You can change it by selecting and dragging it using the mouse. The [Σ Gain] and [Σ Phase] buttons of the **View** panel on the right-hand side are used for changing what is visualized in the diagram: **[Σ Gain]** shows the resulting amplitude response, and **[Σ Phase]** shows the phase response of the sum. You can have both functions enabled at the same time. Using the **[–]** and **[+]** buttons in the **<Bands>** panel, you can increase or decrease the number of bands used one by one.

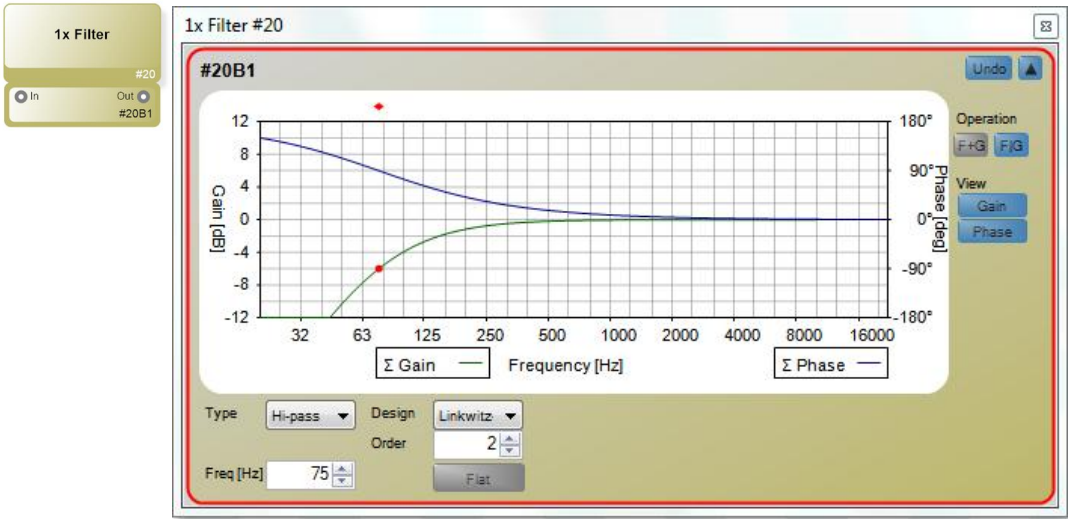
Clicking the **[Flat All]** button toggles the bypass function of the entire function block on and off. In addition, each band features a dedicated **[Flat <x>]** button to enable or disable it separately. When bypass is enabled, the button will be red.

The **<Type>** drop-down menu provides for selecting the filter type of each band. Available types include Peaking, Low-Pass, High-Pass, Notch, All-Pass, Low-Shelf, and High-Shelf. You can view and set the **Gain [dB]** and **Frequency [Hz]** parameters precisely. The value ranges are ±12 dB in 0.1-dB steps and 20-22,000 Hz in 1-Hz steps, respectively. **Bw [Oct]** and **Q Factor** show the filter quality as bandwidth (in octaves) and Q factor, respectively.

➔ The EQ 1..8 block offers the following additional control ports:

Parameter	Control Port	Value Range
Flat All	Logical Input	On/Off

3.3.2 Filter (Audio Function Block: Filter and EQ)



The “Filter” function block provides a configurable hi-pass or low-pass filter and can be used as shelving filter, too. Each block includes up to 64 channels and also handles channel groups.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

The **band** is adjusted either in the diagram using the mouse or by entering the corresponding values in the lower part of the window. Two graphical operating modes can be selected in the right-hand side of the window: In the **[F+G] Mode**, you can configure the frequency and gain settings by moving the band dot horizontally and/or vertically. In the **[F|G] Mode**, you can move the dot only vertically for adjusting the gain while the frequency setting is made using a diamond on top of the diagram. The bandwidth setting (in shelving mode) is visualized by two arrows located on each side of the dot. You can change it by selecting and dragging it using the mouse. The Gain and Phase buttons of the **View** panel on the right-hand side are used for changing what is visualized in the diagram: **Gain** shows the amplitude response and **Phase** the phase response of the filter. You can have both functions enabled at the same time.

The **<Type>** drop-down menu provides for selecting the filter type. Available types include Low-Pass, High-Pass, Low-Shelf, and High-Shelf. The low-pass and high-pass filters are based on either the **Linkwitz-Riley** or the **Butterworth design**. Both designs provide various filter orders (Linkwitz-Riley: 2nd, 4th, 6th, or 8th order; Butterworth: 1st, 2nd, 3rd, 4th, 5th, 6th, 7th, or 8th order). With the shelving bands, you can view and set the **Gain [dB]** and **Frequency [Hz]** parameters precisely. The value ranges are ±12 dB in 0.1-dB steps and 20-22,000 Hz in 1-Hz steps, respectively. **Bw [Oct]** and **Q Factor** show the quality as bandwidth (in octaves) and Q factor, respectively.

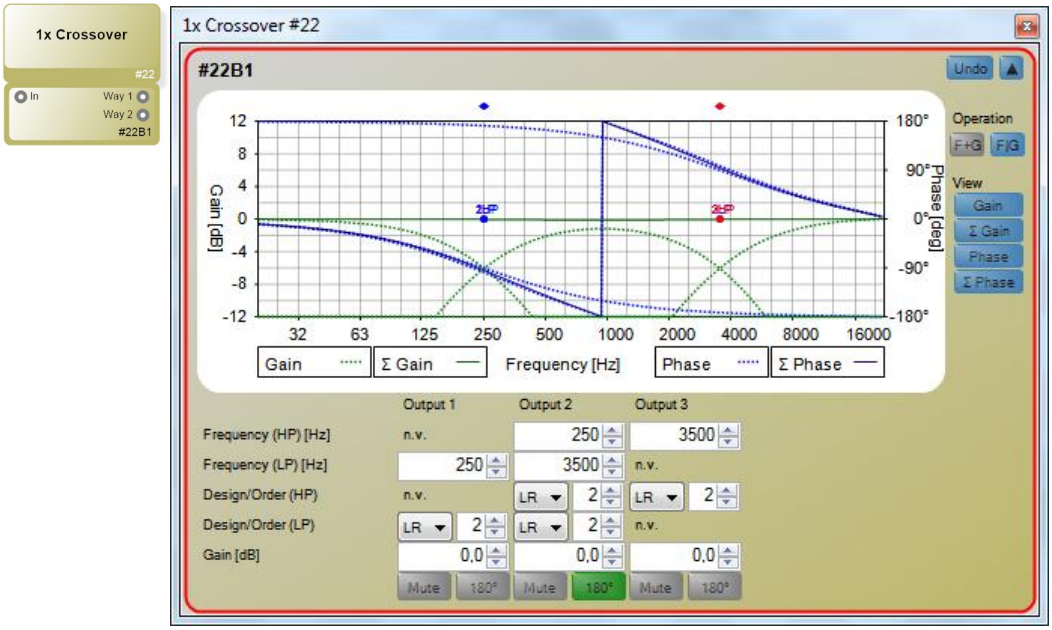
Clicking the **[Flat]** button toggles the bypass function of the entire function block on and off. When bypass is enabled, the button will be red.

➔ The Filter block offers the following additional control ports:

Parameter	Control Port	Value Range

Flat	Logical Input	On/Off

3.3.3 Crossover (2..4) (Audio Function Block: Filter and EQ)



The “Crossover” function block splits the input signal into 2 to 4 output signals using a crossover network. The function provides high-pass and lower-pass filters plus gain for each band. The use of Linkwitz-Riley filters ensures the generation of output signals with smooth amplitude responses. Depending on the selected filter order, specific bands need to be phase-inverted by 180° to be in phase with adjacent bands: Therefore, be sure to enable the Phase button in the View panel for monitoring. Each block includes up to 64 channels and also handles channel groups.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

The crossover-point bands are adjusted either in the diagram using the mouse or by entering the corresponding values in the lower part of the window. Two graphical operating modes can be selected in the right-hand side of the window: In the **[F+G] Mode**, you can configure the frequency and gain settings by moving the band dot horizontally and/or vertically. In the **[F|G] Mode**, you can move the dot only vertically for adjusting the gain while the frequency setting is made using a diamond on top of the diagram. The Gain, Σ Gain, Phase, and Σ Phase buttons of the **View** panel on the right-hand side are used for changing what is visualized in the diagram: **Gain** shows the amplitude response of the individual filter bands; **Σ Gain** the resulting amplitude response of the output bands; **Phase** the phase response of the individual filter bands; and **Σ Phase** the resulting phase response of the output bands. You can have all functions enabled at the same time.

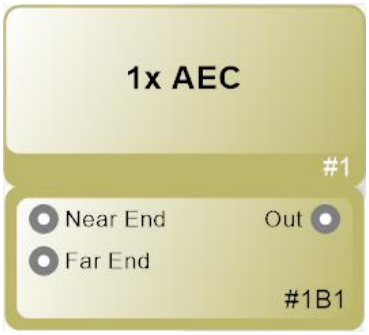
You can view and set the frequencies of the crossover-filter bands precisely using the **Frequency (HP) [Hz]** and **Frequency (LP) [Hz]** parameters. The value range is 20-22,000 Hz in 1-Hz steps. Using the **Design/Order (HP)** and **Design/Order (LP)** settings, you can select the filter design and order: **LR** is a Linkwitz-Riley filter (2nd, 4th, 6th, or 8th order), and **Bw** is a Butterworth filter (1st, 2nd, 3rd, 4th, 5th, 6th, 7th, or 8th order). The output gain of the bands is set using the **Gain [dB]** parameter. The value range is ±12 dB in 0.1-dB steps.

Clicking the **[Mute]** button will mute the respective band. While mute is on, the button will be red.

Clicking the **[180°]** button inverts the polarity of the respective band. While the function is enabled, the button will be green.

➔ The Crossover (2..4) block offers no additional control ports.

3.3.4 AEC – Acoustic Echo Cancellation

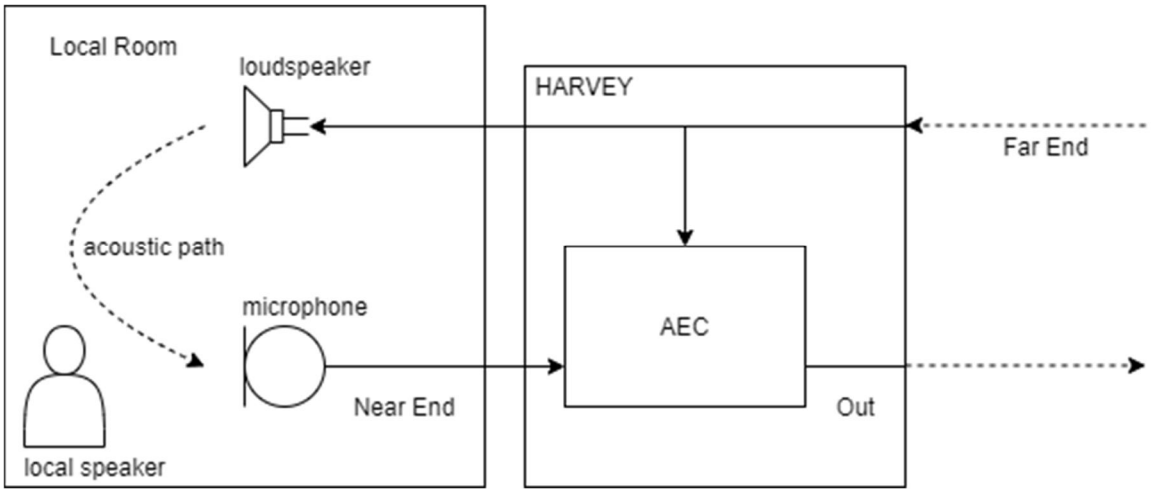


The function block "AEC" ("Acoustic Echo Cancellation") realizes a suppression of acoustic echo, which occurs when a microphone unintentionally picks up the output of a local loudspeaker.

Typical applications for AEC are hands-free communication devices. It is often unavoidable that the microphone of a speaker simultaneously picks up the sound of the loudspeaker over which the other party can be heard. As a result, the other party hears itself with a time delay when speaking (echo).

With the help of an adaptive filter and voice activity detection, the AEC enables hands-free speech without disturbing echo.

The following figure shows the setup schematically:



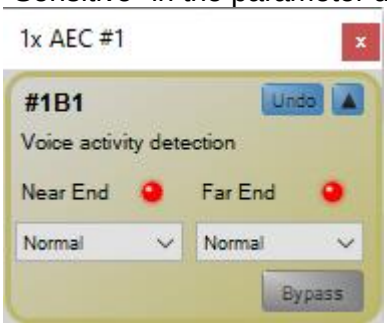
The AEC block offers the following connections in the audio level:

Connection	Type	Description
Near End	Audio Input	Input for the local microphone signal.
Far End	Audio Input	Input for the speech signal received from the far end, which is output locally to the speaker.
Out	Audio Output	The output signal of the AEC, freed from echo, which is to be sent to the far end.

Parameterization

To achieve optimal results, the AEC needs good audio levels of the speakers at the inputs of the block. This can be achieved by good miking in the rooms.

If good leveling is not possible, the voice activity detection of the AEC can be set to "Sensitive" in the parameter dialog for Near End and Far End independently.



The "LED" indicators in the parameter dialog serve as an indicator whether a local (**[Near End]**) or distant **([Far End]**) speaker has been detected. Depending on this, additional attenuation of the output level takes place.

Speaker	**LED-Indicator**	**Meaning**
Near End	Red	No local speaker detected
Near End	Green	Local speaker detected
Far End	Red	No remote speaker detected
Far End	Green	Remote speaker detected

➔ Note: With the "Sensitive" setting, misinterpretation can occur more easily, which is why the "Normal" setting should be preferred.

Example: If Near-End is set to "Sensitive", but the local microphone is in fact well-tuned, a Far-End speaker may be incorrectly detected as Near-End via the echo path.

➔ In most cases, the "Normal" setting should be used. "Sensitive" may be useful when microphones are used at a greater distance (>1m) from the speakers.

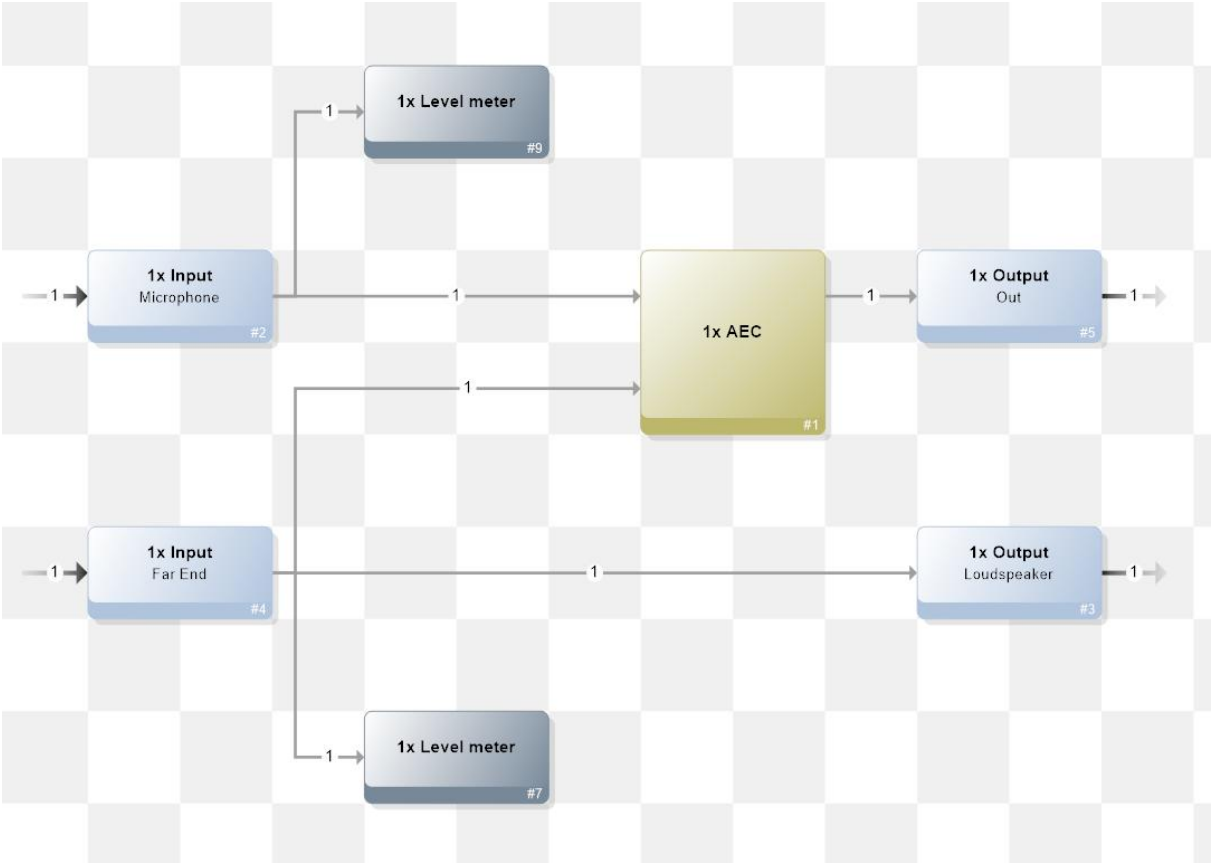
➔ The sensitivity parameters should be adjusted as last resort. Before that, all levels should be checked and set sensibly:
- Is the microphone input gain too high / low?
- Is the loudspeaker in the room too loud / quiet?
- Is there enough signal at the Far End input of the AEC?

Orientation for setting the Near End / Far End voice activity detection:

Scenario	**Sensitivity**
Speaker signal is in the range > -20 dBr	Normal
Speaker signal < -20 dBr	Sensitive

Example scenario

The following example project is intended to provide a guideline for parameterizing the AEC block



The project consists of the following functional blocks:

- Audio input for the local microphone, directly connected to the Near End input of the AEC.
- Audio input for the Far End signal, to which the signal coming from the far end is connected.
- Two level meters to control the input levels
- The AEC block
- An audio output for the output signal of the AEC, this signal must then be transmitted to the far end
- An audio output for the output of the Far End signal

First, the levels of the two inputs are to be set with the help of the level meters. If the available gain is not sufficient, an additional level block can be used. The AEC gives the best results when the level meters of the inputs are around 0dBr for speech.

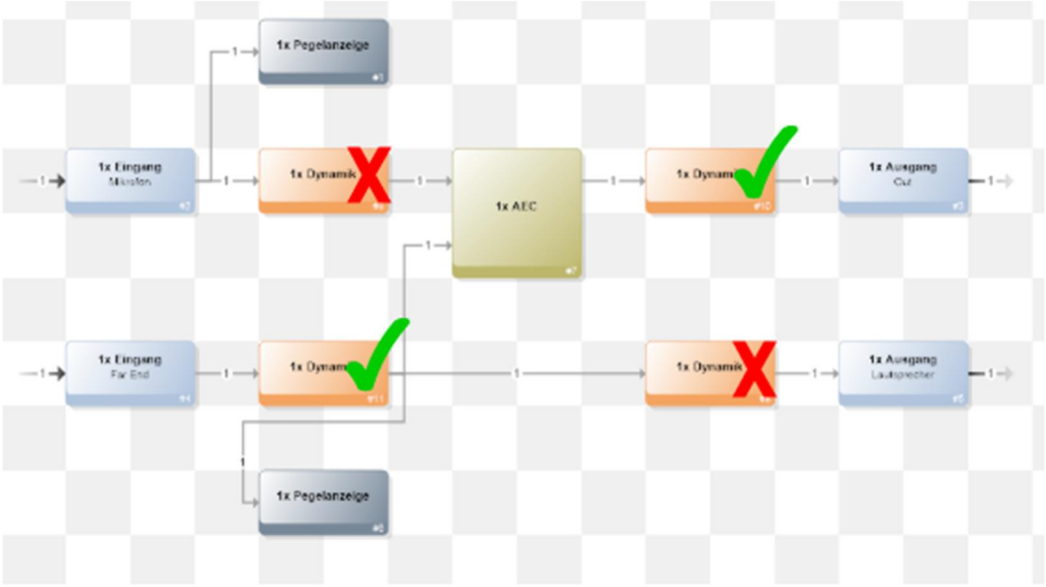
The AEC has a voice activity detection, whose status can be seen via the LEDs in the parameter dialog. Green indicates that a speaker is currently detected.

To further adjust this detection to low speaker levels, you can configure the voice activity detection as “Sensitive”.

Tips:

- Part of Composer are **example projects**. Click on "?" in the menu bar and then on Example projects to open an explorer window. These projects serve as a basis and only need to be adapted to the room and the microphones used.

- Additional **dynamic blocks** can be helpful to keep input levels better under control. However, under no circumstances should dynamics blocks such as dynamics and AGC be placed between the microphone input and the AEC Near End input, or directly before the speaker output. The reason for this is that these dynamics processing will then be seen by the AEC as part of the room and thus worsen the adaptation. Level blocks, on the other hand, can be used at all points without any problems. The example project has been extended with dynamics blocks, with red X are the blocks that should not be used and with green tick the blocks that can be useful:



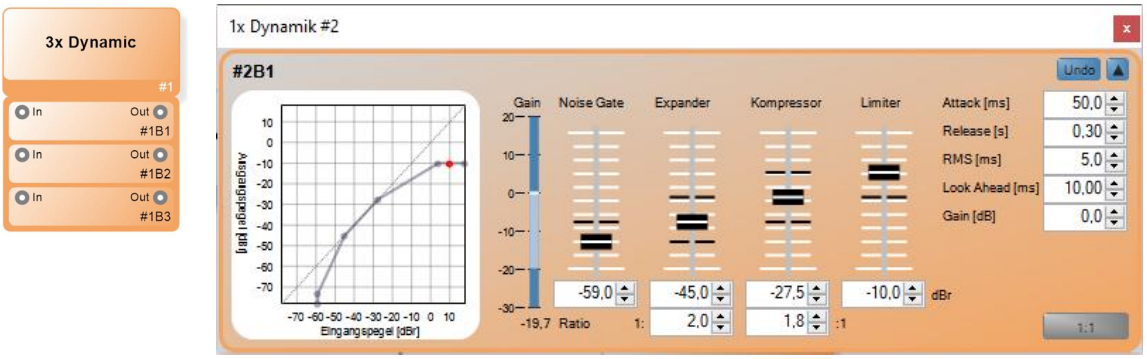
- Microphones in the room should be in the same location and should not be moved while the AEC is in use.
- The near end input of the AEC should not be muted, e.g. a connected microphone should not be switched on/off manually during use.
  - ➔ Noise from open microphones can be eliminated via a **gate** (dynamics block) placed behind the AEC and/or an upstream **Automixer** block.
  - ➔ Caution: An **Automixer** should be set to leave at least one microphone open so that the AEC can continue to adapt. Number of Mics (NoM) and Hold-Time should be set high to avoid irritation of the AEC caused by fast input changes. If the microphones are distributed too far apart, an upstream automixer can still cause irritation for the AEC.
- In general, the AEC will give the best results when the room, speakers and microphones used are optimized for speech.

3.4 Audio Function Blocks: Dynamic

This manual section describes audio function blocks that allow for processing the dynamics of the incoming audio.

➔ Note that Dynamic blocks add to the audio latency in the magnitude of the configured lookahead time.

3.4.1 Dynamic (Audio Function Block: Dynamic)



The “Dynamic” function block combines four dynamic processors. They allow for configuring complex yet comprehensible dynamic processing of the audio signal. The four processors all use the time constants configured in the function block. By setting up the various thresholds purposefully, you can also use the processors separately. This way, you can apply individual dynamic processes with separate time constants by daisy-chaining multiple Dynamic blocks. Each block includes up to 64 channels and also handles channel groups.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

➔ The project-reference level configured in Extras > Options automatically affects the threshold settings in this function block. (Refer to page 16)

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

Using the **fader**, you can set the **threshold** parameter of each dynamics module in steps of 0.1 dB. You can also view and change the threshold in the input fields below the faders. This setting is relative to the configured dBr reference level of the project. The **black marker** on a fader scale show the threshold of the downstream dynamics processors. The expander and compressor provide for adjusting the **ratios** to 1.0:1-20.0:1.

The **diagram** visualizes the ratio between the input level and the output level (i.e. the resulting dynamics curve) as a gray line. Thresholds are marked by **gray dots**.

A **red dot** on the dynamics curve shows the currently measured RMS level of the input signal.

The **blue meter** for gain shows by which value [dB] the input signal is currently amplified or attenuated. If several channels are combined in a group, the values of the first channel of the group are displayed.

Note: For the visualization of the status values, the project must be in online mode and the HARVEY device must have at least version 2.7 firmware installed. If several channels are combined in a group, the values of the first channel of the group are displayed.

The value of **<Attack [ms]>** is the attack time of all dynamics processors (0.1-100 ms).

The value of **<Release [ms]>** is the release time of all dynamics processors (0.01-3 s).

The value of **<RMS [ms]>** is the evaluation-time window. Large time windows ensure smoother dynamics processing while short ones respond faster to peaks (0.1-100 ms).

The value of **<Lookahead [ms]>** is the lookahead time, i.e. an additional delay for performing dynamics processing; however, note that the sidechain loop is still fed with the undelayed input signal. Thanks to this design, the dynamics processor can “look ahead” and thus get some time for triggering their action. The lookahead time can be set within a range of 0.00-1000 ms.

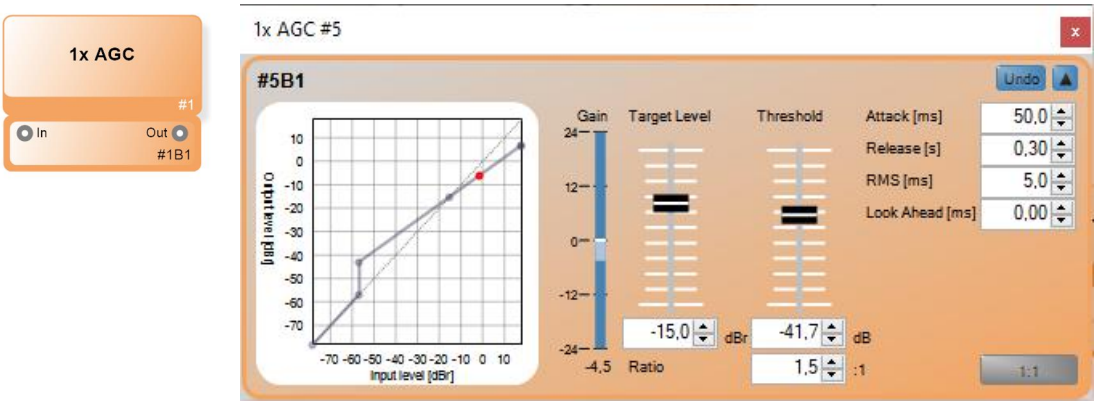
Using the **<Gain [dB]>** control, you can adjust the output level within a range of ±20 dB in 0,1-dB steps.

Clicking the **[1:1]** button toggles the bypass function of the entire function block on and off. When bypass is enabled, the button will be red.

➔ The Dynamic block offers the following additional control ports:

Parameter	**Control Port**	**Value Range**
1:1 (Bypass)	Logical Input	On/Off
Active	Logical Output	On/Off

3.4.2 AGC (Audio Function Block: Dynamic)



The “AGC” (Auto Gain Control) function block provides for straightforward processing used for smoothing audio dynamics. When the input level falls below a defined <threshold>, gain will be increased. This upward compression depends on the configured ratio and the selected threshold. While the signal exceeds the <threshold>, the adjusted compression ratio will be applied; if the level falls below the threshold, the dynamics of the input signal will not be altered <The Target Level> parameter specifies the dBr reference level. While the signal falls below that level, gain increase will be applied; when the <target level> is exceeded, the audio will be downward-compressed applying of the configured ratio. Each block includes up to 64 channels and also handles channel groups.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

➔ The project-reference level configured in Extras > Options automatically affects the threshold settings in this function block. (Refer to page 16)

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

Using the **Threshold** fader, you set in 0.1-dB steps the first threshold where upward compression starts. Gain depends on the **Ratio** setting that can be between 1.0:1-20.0:1. Using the **Target Level** fader, you set in 0.1-dB steps the second threshold where gain increase effectively ends and the downward compression starts. You can also view and change the Threshold and Target Level settings in the input fields below the faders. This setting is relative to the configured dBr reference level of the project.

The **diagram** visualizes the ratio between the input level and the output level (i.e. the resulting dynamics curve) as a gray line. The thresholds are marked by **gray dots**.

A **red dot** on the dynamics curve shows the currently measured RMS level of the input signal.

The **blue meter** for gain shows by which value [dB] the input signal is currently amplified or attenuated.

Note: For the visualization of the status values, the project must be in online mode and the HARVEY device must have at least version 2.7 firmware installed. If several channels are combined in a group, the values of the first channel of the group are displayed.

The value of **<Attack [ms]>** is the attack time of the dynamics processor (0.1-100 ms).

The value of **<Release [ms]>** is the release time of the dynamics processor (0.01-3 s).

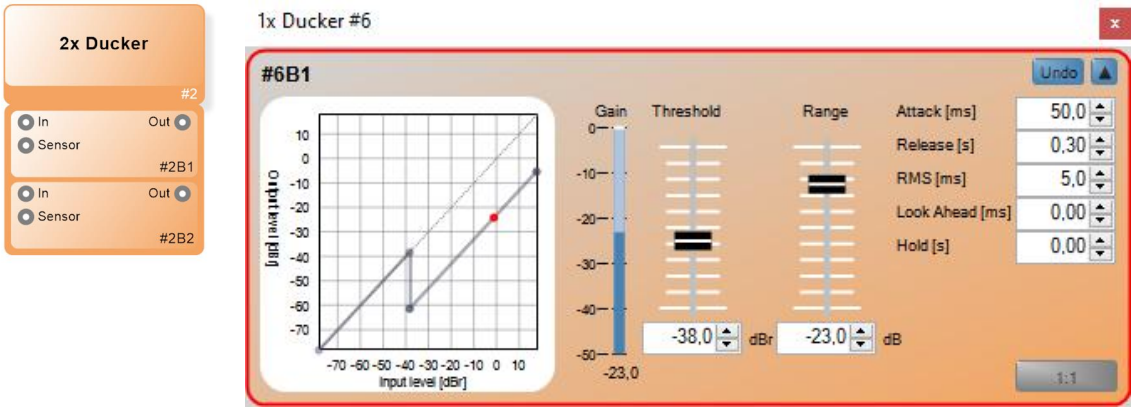
The value of **<RMS [ms]>** is the evaluation-time window. Large time windows ensure smoother dynamics processing while short ones respond faster to peaks (0.1-100 ms).

The value of **<Lookahead [ms]>** is the lookahead time, i.e. an additional delay for performing dynamics processing; however, note that the sidechain loop is still fed with the undelayed input signal. Thanks to this design, the dynamics processor can “look ahead” and thus get some time for triggering their action. The lookahead time can be set within a range of 0.00-1000 ms.

Clicking the **[1:1]** button toggles the bypass function of the entire function block on and off. When bypass is enabled, the button will be red.

➔ The AGC block offers no additional control ports.

3.4.3 Ducker (Audio Function Block: Dynamic)



The “Ducker” function block implements the ducking effect known from radio. In addition to the audio input for the signal to be processed, each channel of this block also has a sensor input. When the level of the sensor input exceeds the <Threshold> setting, the level of the signal present at the audio input will be lowered. You can set the amount of level decrease using the <Range> fader and control it using configurable time constants. Each block includes up to 64 channels and also handles channel groups.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

➔ The reference level configured for the project automatically affects the threshold settings in this function block.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

Using the **Threshold** fader, you select in 0.1-dB steps the level where ducking starts. You can also view and change the threshold in the input fields below the fader. This setting is relative to the configured dBr reference level of the project. The **Range** fader allows for adjusting the amount of level decrease between –100 dB and 0 dB in 0.1-dB steps. You can also view and change the Range setting in the input fields below the faders.

The **diagram** visualizes the ratio between the sensor-input level and decrease in output level as a gray line. Thresholds and post-ducking levels are marked by **gray dots**.

A **red dot** on the dynamics curve shows the currently measured RMS level of the input signal.

The **blue meter** for gain shows by which value [dB] the input signal is currently attenuated.

Note: For the visualization of the status values, the project must be in online mode and the HARVEY device must have at least version 2.7 firmware installed. If several channels are combined in a group, the values of the first channel of the group are displayed.

The value of **<Attack [ms]>** is the ducking attack time (0.1-100 ms).

The value of **<Release [ms]>** is the ducking release time (0.01-3 s).

The value of **<RMS [ms]>** is the evaluation-time window. Large time windows ensure smoother dynamics processing while short ones respond faster to peaks (0.1-100 ms).

The value of **<Lookahead [ms]>** is the lookahead time, i.e. an additional delay for performing dynamics processing; The sidechain loop is, however, still fed with the undelayed input signal. Thanks to this design, the dynamics processor can “look ahead” and thus get some time for triggering their action. The lookahead time can be set within a range of 0.00-1000 ms.

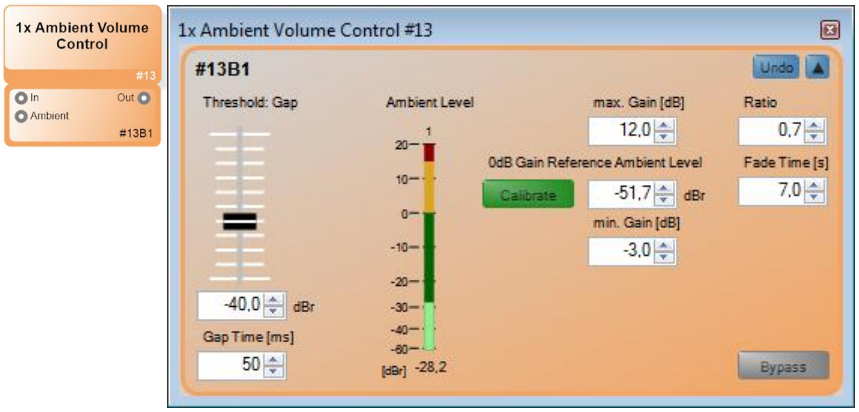
You can extend the ducking by setting a hold time using the **<Hold [ms]>** parameter. The value range is 0.00-10 s.

Clicking the **[1:1]** button toggles the bypass function of the entire function block on and off. When bypass is enabled, the button will be red.

➔ The Ducker block offers the following additional control ports:

Parameter	**Control Port**	**Value Range**
1:1 (Bypass)	Logical Input	On/Off
Active	Logical Output	On/Off

3.4.4 AVC (Audio Function Block: Dynamic)



The AVC (Ambient Volume Control) function block allows for automatically adjusting the program volume (for example, of an announcement) with regard to ambient noise.

In addition to the audio input for the signal to be processed, each channel of this block also has a sensor input where you can connect e.g. a measurement microphone.

During a signal pause or “gap”, when the level of the sensor input exceeds the set 0 dB Gain Reference Ambient Level, the level of the signal present at the audio input will be increased; when the level at the sensor input falls below that reference level, the program level will be reduced.

The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

➔ The reference level configured for the project automatically affects the threshold settings in this function block.

For gap detection in the program signal, you set a **<Threshold [dBr]>** (using the fader) and enter the **<Gap Time [ms]>**. This way, the system detects a gap whenever the program level falls below the threshold longer than the gap time.

The “**Ambient Level”** meter shows the ambient level detected during the last program pause. If several channels are combined in a group, the value of the first channel in the group is displayed. However, grouping channels is not recommended for this block.

Clicking the **[Calibrate]** button applies the detected ambient level as the 0 dB Gain Reference Ambient Level where the program level is neither increased nor reduced.

The **<Max. Gain [dB]>** setting determines the upper threshold that the program level can be increased to in relation to the reference level when the ambience level increases.

The **<Min. Gain [dB]>** setting determines the lower threshold that the program level can be reduced to in relation to the reference level when the ambience level decreases.

The **<Ratio>** parameter sets the ratio of the difference between the ambient and reference levels and the program-gain setting. For example, with a ratio setting of 0.5, when the ambient sound rose by 6 dB, the program level would be increased by just 3 dB.

Using the **<Fade Time [s]>** setting, you can configure how fast the program level adjusts to changed ambience levels.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

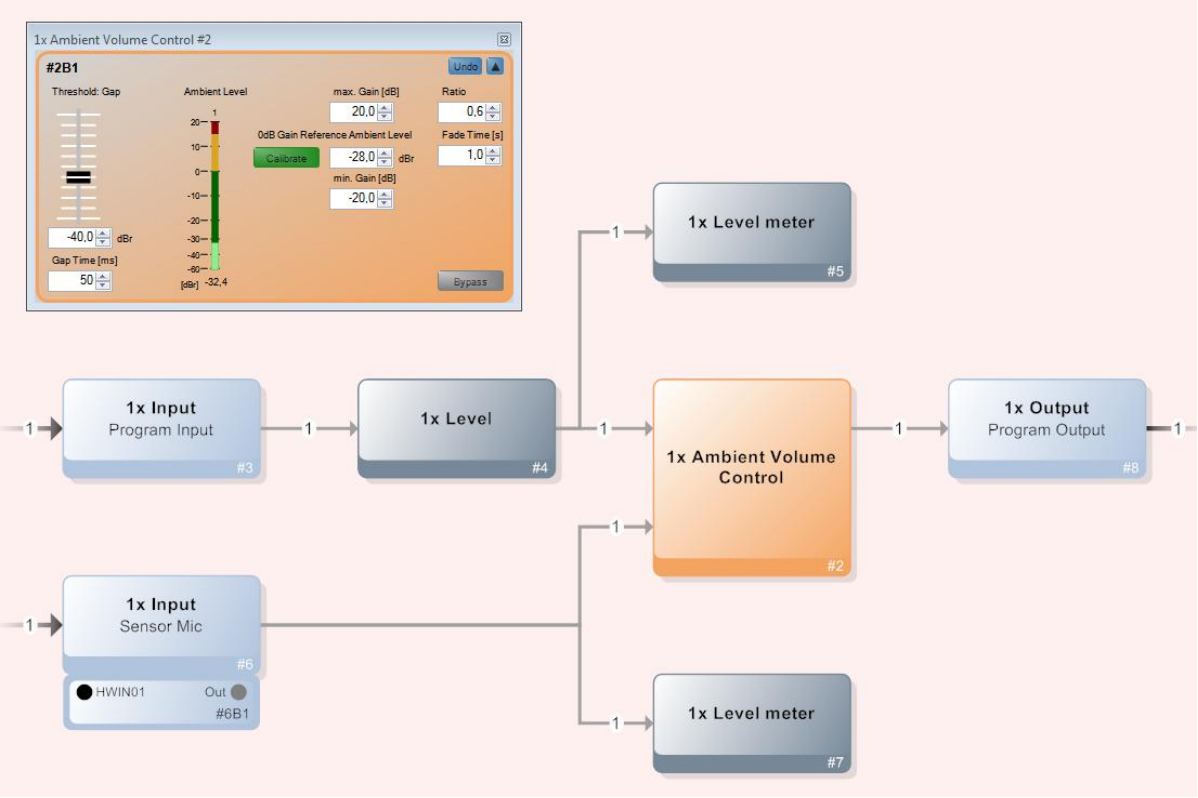
Clicking the **[Bypass]** button toggles the bypass function of the entire function block on and off. When bypass is enabled, the button will be red.

➔ The AVC block offers no additional control ports.

3.4.4.1 Guide to Using the AVC Block

To achieve perfect results when using the AVC function block, you need to configure the best possible settings with regard to the ambient conditions.

This guide provides a sample project to explain how to configure optimum settings for this function block. In this example, we assume that the project in question has been uploaded to a HARVEY device that has an online connection to a computer running HARVEY Composer.



The project includes the following function blocks:

- ➔ An Audio input for the unprocessed announcement signal or program
- ➔ An Audio input receiving this signal of a microphone that measures the ambience volume
- ➔ A Fader used for adjusting the default volume of the announcement signal
- ➔ Two meters for monitoring the program and measurement-microphone levels
- ➔ An AVC function block used for automatically adjusting the program level

➔ An Audio output to send the program signal processed by the AVC to. The speaker system of the venue connects to this audio output.

1. First, using the meters, make sure that the program and measurement-microphone signals have an appropriate level. You can make necessary changes using the Audio Input function blocks or the faders in the program-signal path. If you enable the bypass in the Settings dialog of the AVC function block, the system should reproduce an announcement signal at an appropriate level in the venue.
2. The first step is to configure the gap detection in the Settings dialog of the AVC block.
    a. To do so, an announcement must be replayed.
    b. A Threshold setting of about –40 dBr (at 18 dBr = 0 dBFS) and a Gap Time setting of 50 ms should make a good starting point.
    c. The Ambient Level meter should reflect pauses in the announcement (for example, between sentences). If the meter constantly moves even during the announcements, try reducing the threshold; if the meter shows no sufficient response, increase it. Configure the Gap Time as necessary in a similar fashion.
3. The next step is measuring and configuring the 0-dB reference level, i.e. the ambient level where the program level is neither increased nor reduced.
    a. Enable the bypass function of the AVC function block. Next, start playing the announcement. If necessary, repeat step 1 to make sure that the announcement is played at a volume appropriate for the current ambient volume in the venue. Then, disable the bypass function.
    b. Click the [Calibrate] button to apply the last measured ambience level as the 0-dB reference level.
4. Reasonable gain settings (Max./Min. Gain) depend on the actual conditions of the environment and the resulting variations in ambient volume in the venue:
    a. When calibrating the reference level (step 3), if the ambient volume was normal mode but situations where significantly lower levels occur are still possible, you might want to set the Min. Gain parameter to a value below 0 dB (e.g. –10 dB).
    b. However, if the ambient volume was indeed very low, you might want to consider a Min. Gain setting of 0 dB, thus making sure that the program volume will never be lowered.
    c. Set the Max. Gain parameter so that you get sufficient program amplification when the ambient volume increases.
5. Among other factors, the Ratio setting depends on the position of the measurement microphone: If the microphone is suspended at great height and thus rather far away from the ambient-sound sources on the ground, a Ratio setting of 1 or even greater is recommended; otherwise, select a setting of less than 1.
    a. For testing purposes, you may try to produce different ambient levels inside the venue.
    b. If you feel that AVC does not sufficiently raise the program level when the ambient volume increases, set the Ratio to a higher value. Afterwards, if AVC increases the program level too much, reduce the ratio again.

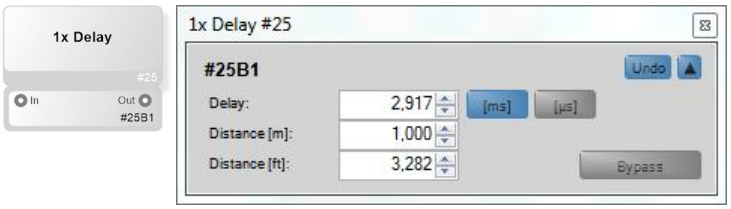
6. Select a high setting (such as 10 seconds) for the Fade Time parameter to prevent perceivable program-level corrections at short-time variations of the ambient volume; however, to achieve faster response, select a lower value.

3.5 Audio Function Blocks: Function Blocks

This section describes function blocks that provide additional functions in the audio domain. These include:

- Delay
- a-b Differential Signal
- Signal Generator

3.5.1 Delay (Audio Function Block: Function)



The “Delay” function block delays the input audio by a number of samples before outputting it. The main application is aligning the latencies of multiple sound sources. The delay time is set as a time or latency value rather than a number of samples.

Each block includes up to 64 channels and also handles channel groups. The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

- ➔ This function block offers ***block options*** through its Settings window. You can increase the step width of the delay by multiplying the sample rate within the block (upsampling). This way, you can switch the step width from the default setting of **20.83 µs** (48,000 Hz) to either **10.64 µs** (96,000 Hz) or even **5.21 µs** (192,000 Hz).
- ➔ When using the upsampling feature, note that the **maximum delay time** decreases from **1000 ms** to **500 ms** or **250 ms**, respectively.
- ➔ In addition, upsampling increases the DSP load, resulting in additional latency. The **Filter Length** in the blog properties allows for adjusting the processing time for upsampling filtering. Low filter settings speed up processing at the cost of higher processing load.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

The value of **<Delay [ms]>** is the delay between the input and the output. The maximum delay time and the step width depend on the settings configured in the block properties. Available step widths include 20.83 µs, 10.64 µs, and 5.21 µs, resulting in maximum delay times of 1000, 500, or 250 milliseconds, respectively. Using the **[ms]** and **[µs]** buttons, you can toggle the unit of time between milliseconds and microseconds.

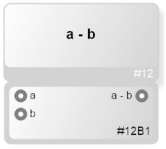
The setting of **<Distance [m]>** parameter determines the resulting sonic latency in meters between the input and the output. The maximum delay time and the step width depend on the settings configured in the block properties. Available step widths include 20.83 µs, 10.64 µs, and 5.21 µs, resulting in maximum propagation speeds of 343, 171.5, or 85.75 m/s, respectively.

The setting of **<Distance [ft]>** parameter determines the resulting sonic latency in feet between the input and the output. The maximum delay time and the step width depend on the settings configured in the block properties. Available step widths include 20.83 µs, 10.64 µs, and 5.21 µs, resulting in maximum propagation speeds of 1125.328, 562.664, or 281.332 ft/s, respectively.

Clicking the **<Bypass>** button toggles the delay on or off. When bypass is enabled, the button will be red.

- ➔ The Delay block offers no additional control ports.

3.5.2 a – b Differential Signal (Audio Function Block: Function)

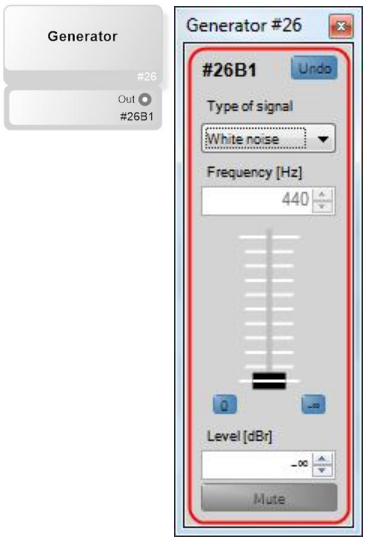


The “a-b” function block generates the difference signal of the two audio signals present at the a and b inputs and outputs. Using this block, you can create the difference signal (or “side signal”) from a L/R stereo source. Difference signals are often used as a basis for pseudo-stereo channels.

Each block includes up to 16 channels. The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

- ➔ The a-b function block provides no configuration options in HARVEY Composer.
- ➔ The a-b block offers no additional control ports.

3.5.3 Sine/Noise Generator (Audio Function Block: Function)



The “Generator” function block includes an audio-signal generator with adjustable output level.

The number of channels automatically adjusts as required. The number of channels automatically adjusts as required. However, you can also set it manually from the Settings window.

➔ The reference level configured for the project automatically affects the displayed generator level (dBr).

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

You can set the type of the generated signal using the **<Signal Type>** drop-down menu. Available types include sine, pink noise, and white noise.

The **<Frequency [Hz]>** parameter is used for setting the sine frequency in 1-Hz steps within a range of 20-22,000 Hz.

Using the **fader**, you can adjust the generator output level in 0.1-dB steps. This setting is relative to the configured dBr reference level of the project. The bottom-stop setting (–∞) is equivalent to muting the channel. Using the blue **[0]** and **[–∞]** buttons allows for configuring a level of 0.0 dBr or –∞, respectively.

You can view the current level setting in the **<Level [dBr]>** field. To change the setting, click into the field and enter the desired value using your keyboard, or use the tiny arrow buttons.

Clicking the **[Mute]** button will mute the generator output. While mute is on, the button will be red.

➔ The Generator block offers the following additional control ports:

Parameter	Control Port	Value Range
Mute	Logical Input	On/Off

4 Control Function Blocks

Serial interfaces, physical control inputs and outputs as well as a web interface are available to control media devices or to operate HARVEY parameters externally. Parallel to the audio level, HARVEY is able to receive or send control commands and, depending on the application, also to link to the audio layer.

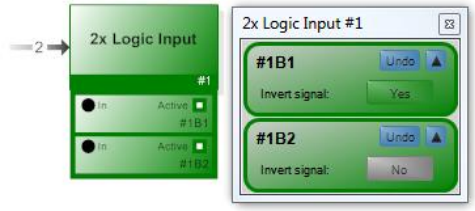
4.1 Control Function Blocks: Inputs and Outputs

On the hardware side, depending on the type, the HARVEY devices offer a different number of logic inputs, logic outputs and voltage inputs.

The logic inputs and logic outputs work with binary logic.

The voltage inputs evaluate voltages. The evaluated voltage levels are converted in the software into level control signals. Thus, by external devices, such as a slider or knob, in the software environment mixed levels can be remotely controlled at function blocks "Level".

4.1.1 Logic Input (Control Function Block: Input / Output)



The function block "Logic Input" transfers the state of the hardware-side logic inputs "Binary Inputs 1-8" of the HARVEY mx.16 or "Control Digital In 1-4" of the HARVEY Pro to the software level. An open switching state is interpreted as "0 / FALSE" and a closed switching state as "1 / TRUE" and provided at the control logic output port. This function block can contain multiple logic inputs simultaneously (up to 8 for HARVEY mx.16 and 4 for HARVEY Pro), with each of these software logic inputs assigned an exclusive HARVEY logic input. Multiple assignments are inadmissible. Each of the hardware ports can only be assigned once. As a result, a maximum of 8 or 4 software logic inputs are possible in one project. Multiple channels can be managed in groups.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

Clicking the **[Invert Signal]** button inverts the binary status at the output port, meaning that 0/FALSE becomes 1/TRUE and vice versa. When signal inversion is enabled, the button label becomes <Yes> and the button color green.

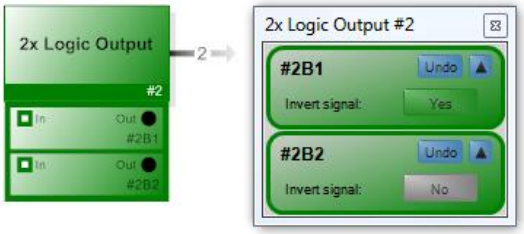
➔ The maximum number of logic inputs differs depending on the device variant and the project type set (see Setting Project Type in the Cockpit on page 13 and Main Window on page 16 and the HARVEY Device Variants on page 139):

Property	**HARVEY Pro**	**HARVEY mx.16[Dante][Cobranet]**
Number of Logic Inputs	4	8

➔ The function block "Logic Input" provides connections in the control layer.

Parameter	**Control Port**	**Value Range**
Output	Logic Output	On/Off

4.1.2 Logic Output (Control Function Block: Input / Output)



The function block "Logic Output" transfers logic signal states from the software level to the hardware-related logic outputs "Relay Outputs 2-4" of the HARVEY mx.16 or "Control Digital Out 1-4" and "Control Relay 1-2" of the HARVEY Pro devices.

The function block may include multiple logic outputs at a time, with each of these software logic outputs being assigned an exclusive HARVEY loigc output. Each of the programmable logic outputs can only be assigned once. As a result, a maximum of 3 or 6 software switching outputs are possible in one project.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

Clicking the **[Invert Signal]** button inverts the logic status at the hardware output, meaning that 0/FALSE becomes 1/TRUE and vice versa. When signal inversion is enabled, the button label becomes <Yes> and the button color green.

➔ With the HARVEY mx.16, the logic output "Relay Output 1" is not available for free programming because it is configured as a fault message relay.

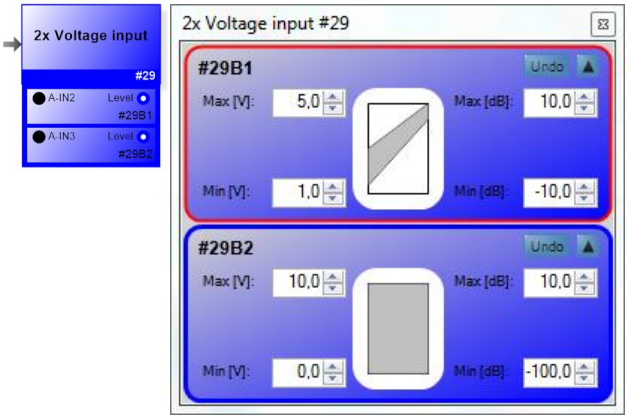
➔ The maximum number of logic outputs and the connection options differ depending on the device variant and the project type set (see Setting project type in the cockpit on page 13 and Main window on page 15 and the summary of device properties on page 108):

Property	**HARVEY Pro**	**HARVEY mx.16[Dante][Cobranet]**
Number of logc outputs	4 transistor outputs 2 relay outputs	No transistor outputs 3 relay outputs
State on activation (1/TRUE) in software Disabled state (0/FALSE) in software	Transistor: Low (Sink) Relay: Switched-on Transistor: High (Source) Relay: Switched-off	Relay: Switched-on Relay: Switched-off

➔ The function block "Logic Output" provides connections in the control layer.

Parameter	**Control Port**	**Value Range**
Input	Logic Input	On/Off

4.1.3 Voltage Input (Control Function Block: Input / Output)



The “Voltage Input” function block converts the voltage level applied to the hardware 0-10 V Control Inputs 1-8 of the HARVEY mx.16 or Control Analog Input 1-2 of the HARVEY Pro to the software level. The interpretation is scalable by the user. This function block can contain multiple voltage inputs at a time, with each of these software voltage inputs assigned an exclusive HARVEY device control input. Multiple assignments are inadmissible. Each of the 8 or 2 voltage inputs can only be assigned once.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

Applied voltage values may be limited to an input voltage range and a control level range for conversion to control levels. With the parameters Min [%] and Max [%] this range is defined in 1% steps between the minimum voltage (0 Volt) and the maximum voltage (mx.16: 10 Volt, Pro: + V). The set levels of the parameters Min [dB] and Max [dB] define the range that is controlled within the defined voltage range. The converted control level is ready to tap on the port (level control output), allowing it to control a channel on the Level function block via the level control input.

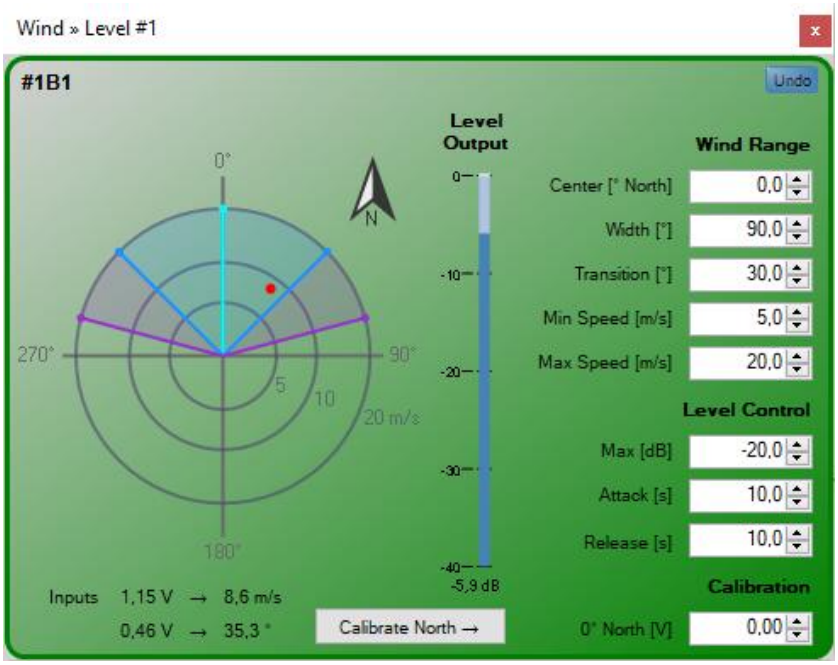
➔ The maximum number of voltage inputs differs depending on the device version and the set project type (see Setting project type in the cockpit on page 13 and Main window on page 16 and the HARVEY Device Variants on page 139):

Property	**HARVEY Pro**	**HARVEY mx.16[Dante][Cobranet]**
Number of Voltage Inputs	2	8
Reference Voltage (Max.)	Ca. 5 V	Ca. 10 V

➔ The function block "Voltage input" provides connections in the control layer:

Parameter	**Control Port**	**Value Range**
Active	Level Output	Defined

4.1.4 Wind » Level



The „Wind » Level" function block enables level control via an anemometer connected to the HARVEY voltage inputs. The latter provides information on wind direction and wind speed.

The intended use case is the lowering of the sound volume, in case of winds from a certain direction, for example for the protection of a residential area.

Wind Range

For the wind direction, the parameters **Center**, **Width** and **Transition** define a range for which a level reduction takes place. Center and Width define the central effective range. For winds within this range, the level is only attenuated depending on the speed.

In the transition area, attenuation is additionally dependent on the angle; the greater the distance to the central effective range, the lower the attenuation. For winds outside the defined range, there is no level attenuation.

With the parameter **Min Speed** the minimum wind speed is defined, for which an attenuation takes place. For wind speeds of **Max Speed** and beyond the attenuation is maximum.

Attention: The direction of 0° refers to winds coming from North. (see *Calibration*). The scaling of wind direction and speed derived from the volt inputs depends on the basic setup. (*see Basic Setup of the Anemometer*)

Level Control

Max declares the maximum level reduction that is applied when the current measured wind speed is greater than Max Speed and the current wind direction is within the central range. **Attack** defines how fast the level reacts to a level reduction and **Release** defines how fast the level reacts to a level raise.

Calibration

The block „Wind » Level" must be calibrated to North. That means: how many volts at the control input "Direction" correspond to a 0° North orientation of the anemometer?

This value must be entered in the input field **0° North [V]**.

Tip: In the simplest case, calibration is performed after mounting the anemometer in the installation. This is done by manually aligning the anemometer to North and holding it in place. Afterwards, the input voltage for 0° North can be directly transferred to the input field in HARVEY Composer in online mode via the **Calibrate North** button.

Status Display

Several status values are displayed in the settings window of the block. For this purpose, a project must be transferred to HARVEY. Displayed are:

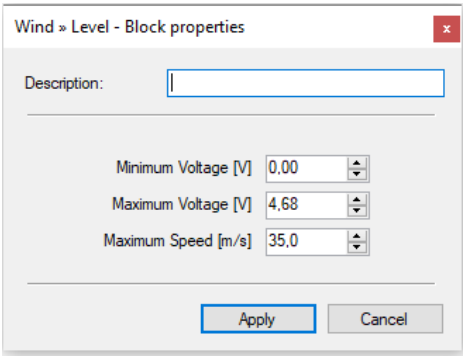
- The wind direction in volts and in degrees (depending on calibration).
- The wind speed in volts and in meters per second
- Within the polar plot the actual measured wind direction and speed is displayed with a red dot.
- The level (or attenuation) applied to the level output of the block is displayed by the meter under **Level Output**.

If the block has no connection to the HARVEY, the status indicators are grayed out.

Basic Setup of the Anemometer

To use an anemometer with HARVEY, it must be ensured that its outputs supply a maximum of 4.68 volts. For this purpose, an adaptation via series resistors may be necessary. The connections of the anemometer are connected to the "Control Analog Inputs" connections of the HARVEY.

Hardware dependent values can be set in the properties dialog of the block.



Minimum voltage indicates the voltage corresponding to a wind direction of 0 degrees and a wind speed of 0 m/s.

Maximum voltage indicates the voltage corresponding to a wind direction of 360 degrees and the maximum wind speed.

Maximum speed indicates the speed measured when the maximum voltage is applied. This value can be found in the data sheet of the anemometer.

4.2 Control Function Blocks: Serial Interfaces

HARVEY devices use RS-232, RS-485, Ethernet and DMX for four different interface types for data transfer and control.

RS-232 and RS-485 are physically available once each.

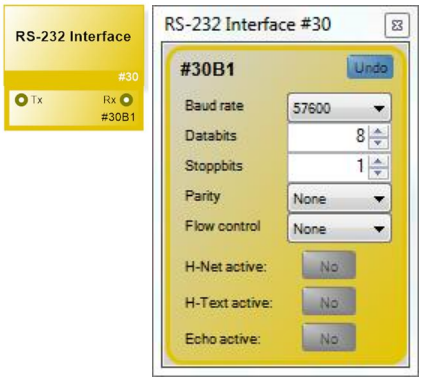
The DMX interface shares the physical connection with the RS-485 interface: RS-485 and DMX cannot be used simultaneously in one HARVEY unit.

The Ethernet interface can be used any number of times as a function block by TCP / IP based TCP.

Particularly interesting is the interface conversion, which for example allows to operate RS-232 or RS-485 capable devices in an Ethernet network, although this is not intended in principle. Likewise, an RS-232 to RS-485 conversion (or vice versa) can be done.

The interface functionality is supplemented by the function blocks "Serial" Logic "and" Logic "Serial", which makes logical functions in the HARVEY software level applicable to the serial interfaces or converts serial commands into logical switching states.

4.2.1 RS-232 Interface (Control Function Block: Serial Interfaces)



The settings of the RS-232 Interface function block are used for configuring the RS-232 interface port of the HARVEY device. It provides one serial RX input and one TX output on the control layer. These ports allow for transmitting serial data packets, which can be used either for configuring logical statuses of control signals (using the Serial » Logic or Logic » Serial blocks) or for triggering the transmission of specific data packets based on logical statuses. In addition, you can interconnect serial interfaces on the control layer for use as an interface converter. Note that you can use just one RS-232 Interface block in each project.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Use the **<Baudrate>** drop-down menu for adjusting the data rate of the RS-232 interface.

The **<Data Bits>** parameter sets the number of bits per data packet.

The **<Stop Bits>** parameter sets the number of stop bits per data packet.

Use the **<Parity>** drop-down menu for selecting whether parity checks will be applied. Settings include None, Even, and Odd.

Using the **<Flow Control>** drop-down menu, you can select the data flow control to be applied. Available settings include None, XON/XOFF, and RTS/CTS.

Use the **<H-Net Active>** button for selecting whether H-Net data will be processed or not. When enabled, the button turns blue.

Use the **<H-Text Active>** button for selecting whether H-Text data will be processed or not. When enabled, the button turns blue.

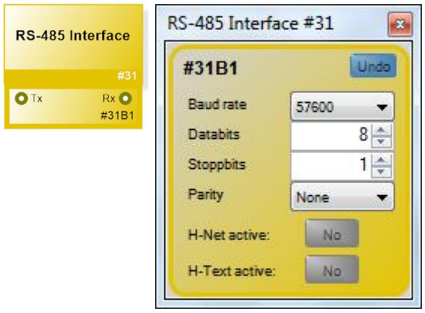
➔ You can enable either H-Net or H-Text.

Use the **<Echo Active>** button for selecting whether H-Text data will be processed or not. When enabled, the button turns blue.

➔ The function block "RS232 Interface" provides connections in the control layer:

Parameter	Control Port	Value Range
RX Port	Serial Input	Data Packet
TX Port	Serial Output	Data Packet

4.2.2 RS-485 Interface (Control Function Block: Serial Interfaces)



The settings of the RS-485 Interface function block are used for configuring the RS-485 interface port of the HARVEY device. It provides one serial RX input and one TX output on the control layer. These ports allow for transmitting serial data packets, which can be used either for configuring logical statuses of control signals (using the Serial » Logic or Logic » Serial blocks) or for triggering the transmission of specific data packets based on logical statuses. In addition, you can interconnect serial interfaces on the control layer for use as an interface converter. Note that you can use just one RS-485 Interface block in each project.

The RS-485 hardware port of the HARVEY device can be used as either RS-485 or DMX port. Therefore, you cannot use the RS-485 Interface and DMX function blocks at the same time.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Use the **<Baudrate>** drop-down menu for adjusting the data rate of the RS-485 interface.

The **<Data Bits>** parameter sets the number of bits per data packet.

The **<Stop Bits>** parameter sets the number of stop bits per data packet.

Use the **<Parity>** drop-down menu for selecting whether parity checks will be applied. Settings include None, Even, and Odd.

Use the **<H-Net Active>** button for selecting whether H-Net data will be processed or not. When enabled, the button turns blue.

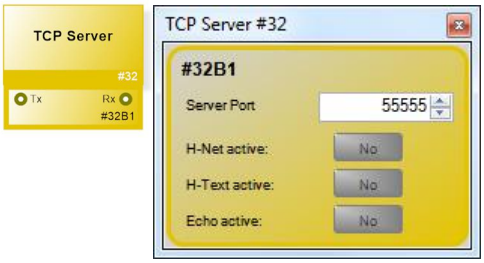
Use the **<H-Text Active>** button for selecting whether H-Text data will be processed or not. When enabled, the button turns blue.

➔ You can enable either H-Net or H-Text.

➔ The function block "RS485 Interface" provides connections in the control layer:

Parameter	Control Port	Value Range
RX Port	Serial Input	Data Packet
TX Port	Serial Output	Data Packet

4.2.3 TCP Server (Control Function Block: Serial Interfaces)



The TCP Server function block activates a HARVEY device as TCP server, so that external (Client) devices can control the HARVEY device via an Ethernet network. The function block provides one serial RX input and one TX output on the control layer. These ports allow for transmitting serial data packets, which can be used either for configuring logical statuses of control signals (using the Serial » Logic or Logic » Serial blocks) or for triggering the transmission of specific data packets based on logical statuses. In addition, you can interconnect serial interfaces on the control layer for use as an interface converter.

You can use any number of UDP Client function blocks.

Clicking the **Undo** button will undo all changes made since the Settings window was opened.

In the **Target Port** field, you can enter a value between 2000 and 59999 to set the port used for addressing the TCP server.

Use the **<H-Net Active>** button for selecting whether H-Net data will be processed or not. When enabled (Yes), the button turns blue.

Use the **<H-Text Active>** button for selecting whether H-Text data will be processed or not. When enabled, the button turns blue.

➔ You can enable either H-Net or H-Text.

Use the **<Echo Active>** button for selecting whether H-Text data will be processed or not. When enabled, the button turns blue.

➔ The function block "TCP Server" provides connections in the control layer:

Parameter	Control Port	Value Range
RX Port	Serial Input	Data Packet
TX Port	Serial Output	Data Packet

4.2.4 UDP Client (Control Function Block: Serial Interfaces)

The UDP Client function block allows for controlling UDP-enabled external (server) devices on an Ethernet network using the HARVEY device. The function block provides one serial RX input and one TX output on the control layer. These ports allow for transmitting serial data packets, which can be used either for configuring logical statuses of control signals (using the Serial » Logic or Logic » Serial blocks) or for triggering the transmission of specific data packets based on logical statuses. In addition, you can interconnect serial interfaces on the control layer for use as an interface converter.

You can use any number of UDP Client function blocks.

Clicking the **Undo** button will undo all changes made since the Settings window was opened.

In the **Target Port** field, you can enter a value between 2000 and 59999 to set the port used for addressing the device to be controlled (i.e. the server).

Next, enter the IP address of the device to be controlled into the **Target IP Address** field.

➔ The function block "UDP Client" provides connections in the control layer:

Parameter	Control Port	Value Range
RX Port	Serial Input	Data Packet
TX Port	Serial Output	Data Packet

4.3 Control Function Blocks: Logic

The “Negator”, “RS Flipflop”, “T-FlipFlop” and “Level Up Down” function blocks add more functions for logical control.

4.3.1 Negator (Control Function Block: Logic)

The “Negator” function block inverts the logical signal present at the input, meaning that it outputs the opposite status: 0/FALSE becomes 1/TRUE and vice versa.

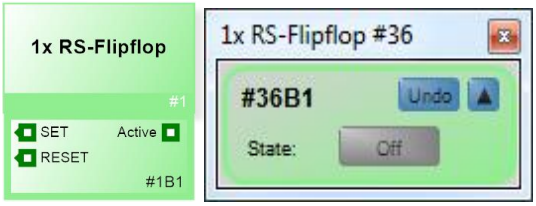
Each block includes up to 64 channels. The number of channels automatically adjusts as required.

The “Negator” function block provides no configuration options in HARVEY Composer.

➔ The function block "Negator" provides connections in the control layer:

Parameter	**Control Port**	**Value Range**	**Status (1)**
Input	Logical Input	On/Off	Negated status that was returned to Output.
Output	Logical Output	On/Off (Inverted)	Can receives Status that is being returned by attached block.

4.3.2 RS Flipflop (Control Function Block: Logic)



The “RS Flipflop” function block implements an RS flip-flop: Momentarily applying the 1/TRUE status to the SET input results in 1/TRUE as the stable output status while momentarily applying 1/TRUE to the RESET input resets the output status to 0/TRUE. If 1/TRUE is present at both inputs at the same time, a random output status will be set.

The block includes up to 64 channels. The number of channels automatically adjusts as required.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

Using the **<State>** button, you can set the output of the RS Flipflop block to 1/TRUE for testing purposes. When enabled, the button turns green.

➔ The function block "RS-FlipFlop" provides connections in the control layer:

Parameter	**Control Port**	**Value Range**	**Status (1)**
Set	Logical Input	On/Off (Evaluation)	Flipflop State
Reset	Logical Input	On/Off (Evaluation)	Inverted FF State
Output	Logical Output	On/Off (Evaluation)	-/-

(1) The Logical Inputs return the Flipflop state as status (plain or inverted), e.g. to toggle color switching on attached RC4 or RC12 buttons.

4.3.3 T-Flipflop (Control Function Block: Logic)

The T-Flipflop function block implements a toggle flip-flop: Momentarily applying a signal to the toggle input inverts the output status.

The block includes up to 64 channels. The number of channels automatically adjusts as required.

Clicking the **Undo** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

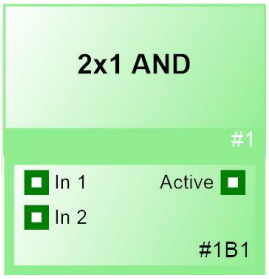
Using the **State** button, you can set the output of the T-Flipflop block to 1/TRUE for testing purposes. When enabled, the button turns green.

➔ The function block "T-FlipFlop" provides connections in the control layer:

Parameter	**Control Port**	**Value Range**	**Status**
Toggle	Logic Input	On/Off (Evaluation)	Flipflop State
Output	Logic Output	On/Off (Evaluation)	-/-

(1) The Toggle Input returns the Flipflop State as status, e.g. to toggle color switching on attached RC4 or RC12 buttons.

4.3.4 AND (Control Function Block - Logic)



The "AND" function block is a logic gate that implements a logical conjunction. If a "1/TRUE" is present at all inputs, "1/TRUE" is also present at the output.

The block can have up to 64 inputs. Any input that is not connected is automatically treated as "0/FALSE".

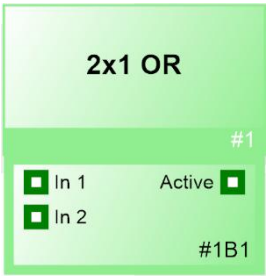
The number of input ports adjusts automatically when a cable bundle is pulled onto the block. However, the number can also be defined manually via the block properties.

The function block "AND" does not provide function block settings in the HAREY Composer software.

➔ The function block „AND“ provides connections in the control layer:

Parameter	**Control port**	**Value range**
In #	Logic Input	On/Off (Evaluation)
Active	Logic Output	On/Off

4.3.5 OR (Control Function Block - Logic)



The "OR" function block is a logic gate that implements a logical disjunction. If a "1/TRUE" is present at at least one of the inputs, "1/TRUE" is also present at the output.

The block can have up to 64 inputs. Any input that is not connected is automatically treated as "0/FALSE".

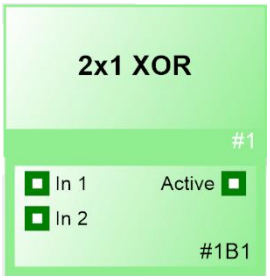
The number of input ports adjusts automatically when a cable bundle is pulled onto the block. However, the number can also be defined manually via the block properties.

The function block "OR" does not provide function block settings in the HAREY Composer software.

➔ The function block „OR“ provides connections in the control layer:

Parameter	Control port	Value range
In #	Logic Input	On/Off (Evaluation)
Active	Logic Output	On/Off

4.3.6 XOR (Control Function Block - Logic)



The "XOR" function block is a logic gate that implements a logical excluding disjunction. If a "1/TRUE" is present at only of the inputs, "1/TRUE" is also present at the output.

The block can have up to 64 inputs. Any input that is not connected is automatically treated as "0/FALSE".

The number of input ports adjusts automatically when a cable bundle is pulled onto the block. However, the number can also be defined manually via the block properties.

The function block "XOR" does not provide function block settings in the HAREY Composer software.

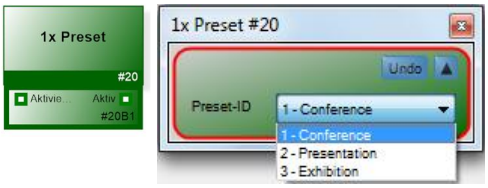
➔ The function block „XOR“ provides connections in the control layer:

Parameter	**Control port**	**Value range**
In #	Logic Input	On/Off (Evaluation)
Active	Logic Output	On/Off

4.4 Control Function Blocks: Function Blocks

The Preset function block described in this section allows for automated parameter recall when specific conditions are met. This way, using external switches, you can trigger specific operating statuses that implement different applications of an installation.

4.4.1 Preset (Control Function Block: Function Blocks)



When the Preset block receives 1/TRUE at its control input, it loads a preset configured for the current project. This way, the block allows for loading stored parameter sets. The block also has a control output which represents the active state of the selected preset. That state will also be signaled if the preset is being called via other mechanisms (e.g. H-Text).

The block includes up to 64 channels. The number of channels automatically adjusts as required.

You can use this function block for switching between several application scenarios or even for automatically changing parameters when specific statuses occur.

Using the serial interfaces or binary inputs, you can enable various statuses and parameters on multiple HARVEY mx.16 units.

In addition, messages can be triggered using H-Net and H-Text.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

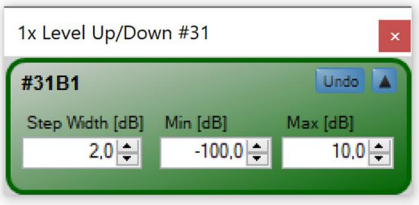
Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

Using the **<Preset ID>** drop-down menu, you select a preset that will be applied when 1/TRUE is present at the input.

➔ The function block "Preset" provides connections in the control layer:

Parameter	Control Port	Value Range
Input	Logical Input	On/Off (Evaluation)
Output	Logical Output	On/Off

4.4.2 Level Up/Down



The „Level Up/Down“ Block is a control block for changing level controls step by step. Activation at the logic input "Up" leads to an increase in the level by the set increment. Accordingly, activation at the logic input "Down" leads to a reduction in the level by the increment.

Activation at the logic input means a rising edge, i.e. a transition from "0 / FALSE" to "1 / TRUE". The logic inputs do not react to falling edges.

The block can contain up to 64 channels. The number of channels required adjusts automatically if necessary.

Clicking the **Undo** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

With **<Step Width [dB]>** a value is set by which the connected level is changed if a short-term activation is applied to one of the logic inputs. A minimum of 0.1dB and a maximum of 110dB can be set.

<Min [dB]> is used to set the minimum value of the level output. If the value is changed and a connected level block is below **<Min [dB]>**, the value of the level block is adjusted.

<Max [dB]> is used to set the maximum value of the level output. If the value is changed and a connected level block is above **<Max [dB]>**, the value of the level block is adjusted.

➔ The block "Level Up/Down" provides connections in the control layer:

Parameter	Control Port	Value Range
Up	Logic Input	On/Off
Down	Logic Input	On/Off
Level	Level Output *not exlusive(1)*	Defined by Min/Max

(1) The connected level is not exclusively controlled by the "Level Up/Down" block and can still be parameterized via other ways. (e.g. H-Text or Presets)

Note: The Level Up/Down block does not have audio connectors. Rather, it must be connected to an audio block that has a level control input.

4.4.3 Zone Control

The "Zone control" block forwards control signals to one of several zones depending on the zone selection. The number of signals and zones can be set in the block properties.

In the application, the block serves to enable an upstream controller (e.g. RC12) in a simple way to alternatively control different zones. It is up to the user how individual zones are realized in the project. HARVEY Composer does not provide a fixed structure for this. The example project below represents zones as alternatively controllable rooms with similar control structure, but it is also possible to realize different control scenarios per zone.

The selection of the zone to be controlled is done either by directly setting the internal parameter via the block settings, by preset or by activating one of the control inputs in the section "Selection".

Activation of the logic input means a rising edge, i.e., a transition from "0/FALSE" to "1/TRUE". The selection inputs of the block do not react to falling edges.

The block can serve up to 64 connections in total combining all zones. This implicitly results in the limitation of the number of zones and signals per zone, for example 8x8 or 4x16.

Dynamic Signal Typing

Signal inputs and outputs are initially not bound to a specific signal type. That state is indicated by a grey star port symbol. Logic, level or DMX connections can be connected.

As soon as a control connection is connected, the type of the signal input as well as the associated signal outputs change to match the type of the connected signal wire.

Example: When a *logic* wire is connected to input "Signal 2", the "Signal 2" outputs of all zones automatically change to *logic* outputs.

A zone control block can forward different signal types. For example, "Signal 1" may be a logic signal while "Signal 2" is a level control signal.

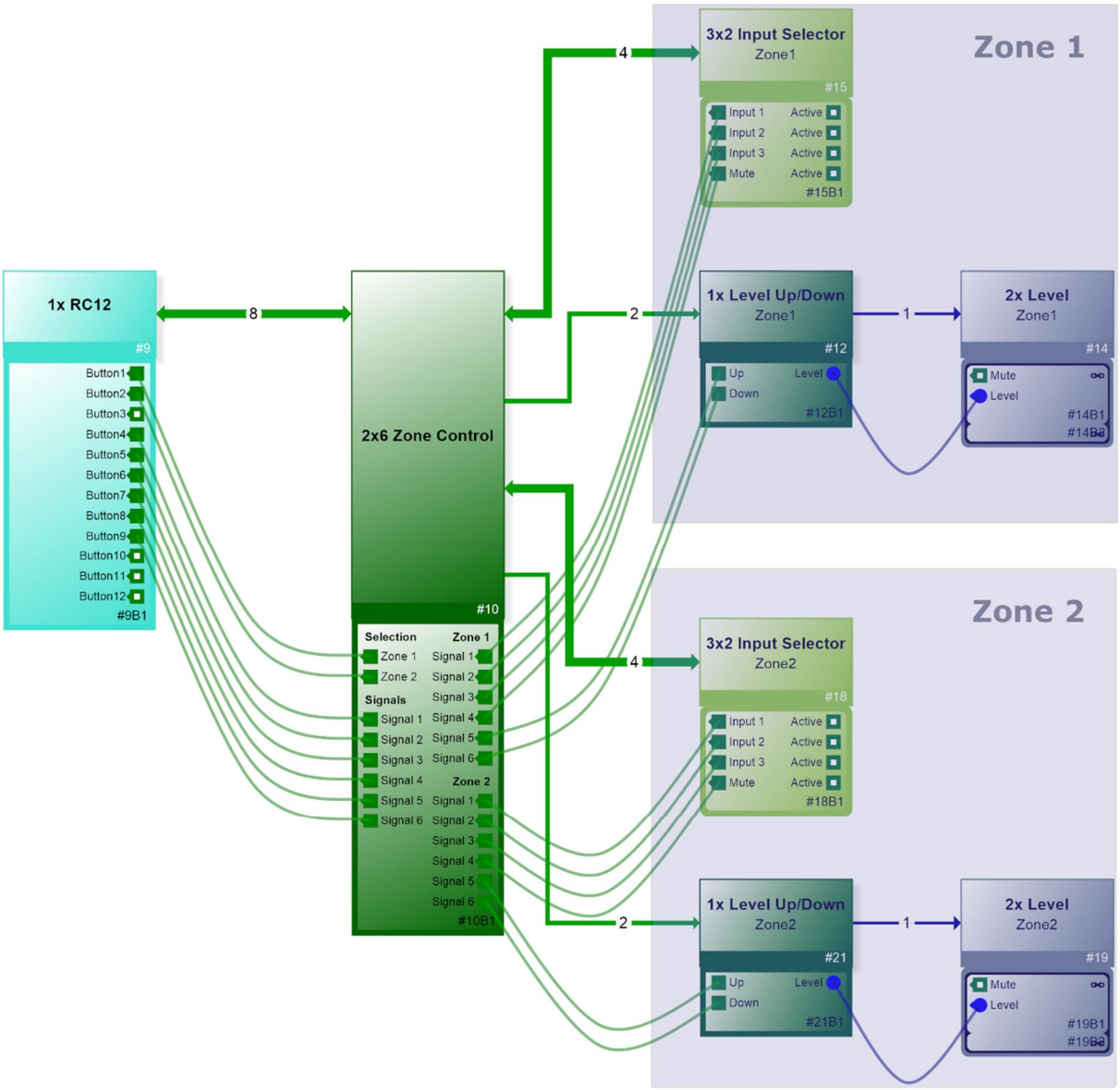
➔ The „Zone Control“ Block offers the following additional control ports:

	Parameter	Control Port	Value Range	Function	Status (1)
Per Zone #	Zone #	Logical Input	On/Off	Select Zone # in Zone Control	Is Zone # selected?
	Signal 1..N	Logical, Level or DMX Output	Varies depending on port type	Passthrough of signals 1..N if Zone # is selected	Can receive status from an attached sink
	Signal 1..N	Logical, Level or DMX Input	Varies depending on port type	Attached signal will be forwarded to the corresponding output of the selected zone.(2)	Returns Status received by the corresponding signal output in the selected zone.

(1) The inputs the current parameter state as status to the attached block (e.g. to trigger color switching on an attached RC4 or RC12).

(2) Note: When a zone is selected, the signal levels present at the signal inputs are not passed through to the newly selected zone until they are changed.

(3) Whether a signal output takes exclusive control of a connected parameter depends on the behavior of the connected source at the corresponding signal input.



The figure shows an example of the integration of the "Zone Control" block in a project with two zones. The connected RC12 thus gains the following functionality:

- ➔ After selecting the zone to be controlled via pushbutton 1 or 2, the RC12 pushbuttons 4 to 9 act as if they were directly connected to the blocks in the selected zone.
- ➔ Perform Source selection in the selected zone via buttons 4 - 7
- ➔ Perfom Level control in the selected zone via buttons 8 and 9
- ➔ Display of the feedback statuses via the parameterized button colors

Note: Audio connections are hidden in the figure for better overview.

4.5 Control Function Blocks: Serial

The function blocks described in this section allow for evaluating or triggering serial data packets.

4.5.1 Serial » Logic (Control Function Block: Serial)

When receiving a specific serial data packet, the “Serial » Logic” function block momentarily outputs the 1/TRUE status. To achieve a stable logical status, you need to insert a downstream “RS-Flipflop” function block. The Serial » Logic block includes a serial RX input for data-packet reception on the control layer.

The block includes up to 64 channels. The number of channels automatically adjusts as required.

Multiple channels share the same serial RX input.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel in the Settings window.

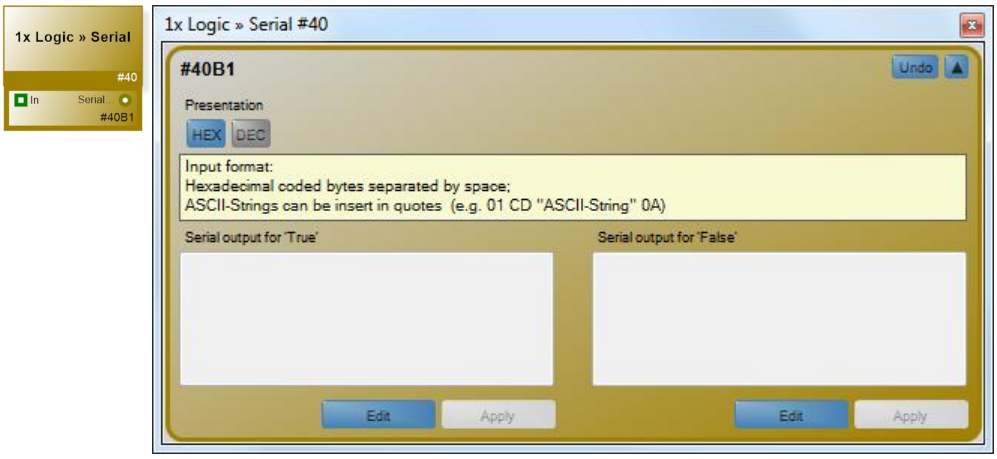
The **[HEX]** and **[DEC]** buttons allow for toggling the data-packet byte view and input between hex and decimal. To enter ASCII strings, put them in quotes. For example: 01 CD "ASCII string" 0A.

You can view and enter the data packet that will trigger the new logical status at the control output in the **<Serial Input>** field. Use spaces as a delimiter when entering the encoded bytes. Clicking the **[Edit]** button below the input field enables the input mode. Click the input field and enter your characters using the mouse. When doing so, blue **[Apply]** and **[Cancel]** buttons will be displayed. Using these buttons, you can apply your changes or cancel the action; in both cases, you exit the input mode.

➔ The function block " Serial » Logic " provides connections in the control layer:

Parameter	Control Port	Value Range
RX	Serial Input	Data Packet
Output	Logical Output	On/Off (Evaluation)

4.5.2 Logic » Serial (Control Function Block: Serial)



The Logic » Serial function block evaluates the control-input status and sends a previously defined serial data packet when TRUE or FALSE are received. It provides one serial TX output on the control layer used for routing the triggered data packets to a serial interface.

The block includes up to 64 channels. The number of channels automatically adjusts as required. Multiple channels share the same serial TX output.

Clicking the **[Undo]** button will undo all changes made since the Settings window was opened.

Using the **[^]** and **[v]** buttons, you can show or hide the parameters of a channel group in the Settings window.

The **[HEX]** and **[DEC]** buttons allow for toggling the data-packet byte view and input between hex and decimal. To enter ASCII strings, put them in quotes. For example: 01 CD "ASCII string" 0A.

The data packets for each signal state are entered into the **<Serial Output for 'True'>** and **<Serial Output for 'False'>** fields, respectively. Use spaces as a delimiter when entering the encoded bytes. Clicking the **[Edit]** button below the input fields enables the input mode. Click the input field and enter your characters using the mouse. When doing so, blue **[Apply]** and **[Cancel]** buttons will be displayed. Using these buttons, you can apply your changes or cancel the action; in both cases, you exit the input mode.

➔ The function block " Logic » Serial " provides connections in the control layer:

Parameter	Control Port	Value Range
Input	Logical Input	On/Off (Evaluation)
TX	Serial Output	Data Packet

4.6 Control Function Blocks: DMX

HARVEY devices can be integrated into a DMX network via the RS-485 interface.

Through the DMX integration, HARVEY devices can control other DMX devices and, for example, use presets to implement different lighting scenarios. In this case, soft transitions of the value changes (dimming, fading) per preset can be configured.

HARVEY devices can also participate as a client on the DMX network and be controlled by other DMX controllers. In addition, DMX allows direct control of level blocks in the HARVEY project to control the volume. However, it is also possible to trigger any control events in the HARVEY project in a threshold-controlled manner. This ranges from simple preset calls to the triggering of messages to other connected devices. The HARVEY device thus becomes the gateway and allows almost any control of third-party devices via DMX.

Function blocks in this section are used to configure the DMX functionality of the device.

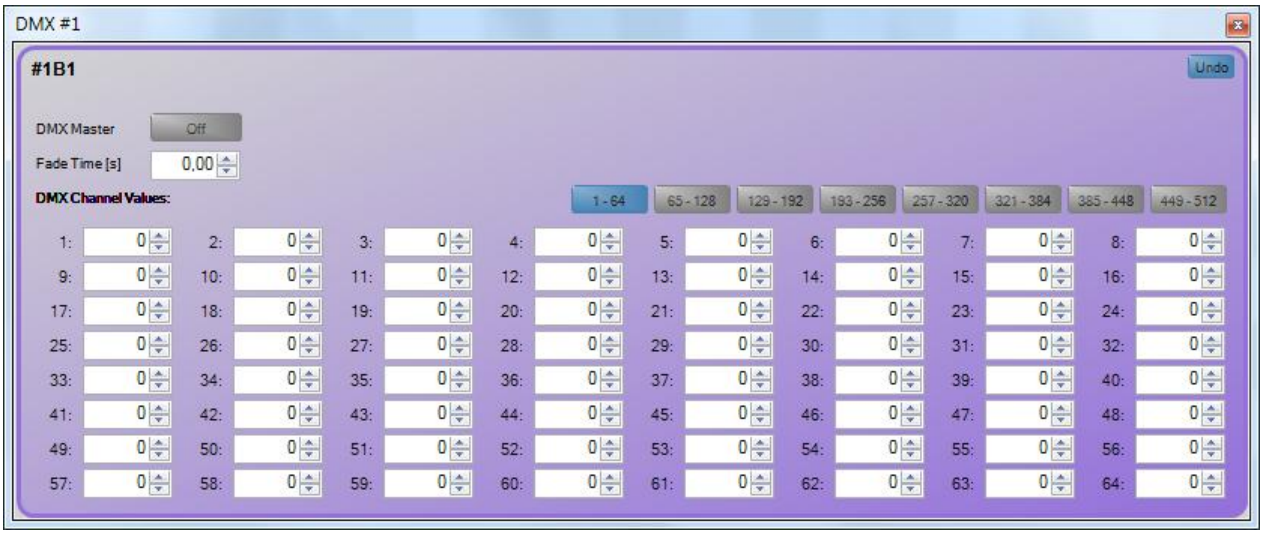
4.6.1 DMX (Control Function Block: DMX)

The function block "DMX" reserves and activates the RS-485 interface for operation with digital DMX control signals. HARVEY can basically act as a sender (master) or receiver (slave) on the DMX bus.

- ➔ The hardware-side RS-485 interface on the HARVEY device can be used either as an RS 485 interface or as a DMX interface. The function blocks "RS-485 interface" and "DMX" cannot be used simultaneously. Only one of the two blocks can be used at a time.
- ➔ Only one transmitter may be active on a DMX bus. HARVEY may only be activated as a transmitter if no other DMX controller is active on the bus.

DMX parameter dialog

The parameter dialog is opened by double-clicking on the DMX block icon in the worksheet.



Clicking the **[Undo]** button in the Setting window will undo all changes made since the window was opened.

The **<DMX Master>** button activates HARVEY to be the DMX controller on the bus. When enabled, the button turns green. If disabled, HARVEY mx.16 is capable of receiving DMX channel values from another DMX controller.

The DMX block features 512 **<DMX Channel Values>** which appear in groups of 64 channels in the Settings window. The groups are only used to keep the interface small: No matter which group is selected in an open Settings window the block will always contain all 512 channel values.

DMX Channel values may be stored in presets. The channels which are to be stored in a preset have to be selected within that preset's parameter tree (see Presets, 2.6.2). The **<Fade Time>** parameter may be used to set up smooth fading of DMX values upon a preset call over the configured time. If the **<Fade Time>** parameter is being added to the preset as well it can be used to set specific fade times for each preset.

➔ The function block "DMX" provides connections in the control layer:

Parameter	Steueranschluss	Bereich
Input DMX1..16	DMX Input	0..255
Output DMX	DMX Output	0..255

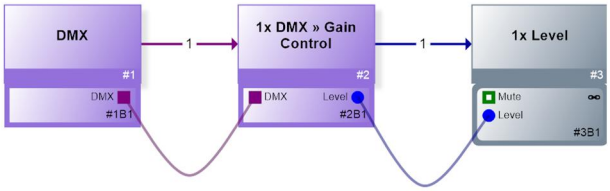
At the output port of the DMX block, the values of all DMX channels are transmitted to attached control blocks ("DMX » Gain Control" on page 120, "DMX » Logic" on page 121).

The input ports of the DMX block can control the DMX channels 1-16 of RC4 remote control blocks ("RC4-DMX").

4.6.2 DMX » Gain Control (Control Function Block: DMX)

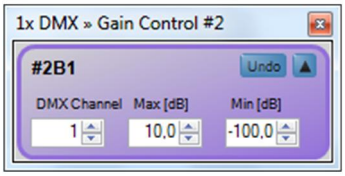
The “DMX » Gain Control” function block adapts DMX channel values to level gain parameters. It therefor allows to for Level blocks to be controlled by external DMX controllers.

The “DMX » Gain Control” block input must be connected to a DMX Source (DMX block, see 4.6.1). The output must be connected to level input of the “Level” block which is to be controlled.



Note: The adaption uses the same scaling as the “Level” block Setting window in Composer giving a fine-grained control over the region around 0 dB. As a result, common linear hardware DMX faders obtain a characteristic as is known from professional audio mixing consoles.

“DMX » Gain Control” parameter dialog:



The parameter dialog allows a channel-by-channel configuration of the following parameters:

Clicking the **[Undo]** button in the Setting window will undo all changes made since the window was opened.

Using the **[∧]** and **[v]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

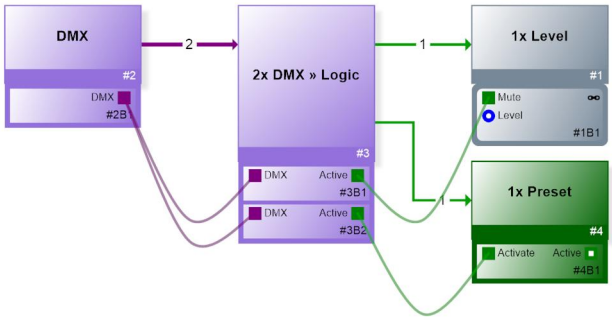
The **<DMX Channel>** field controls which DMX channel value is to be used for the gain control adaption. DMX Channel accepts values from 1 to 512.

<Max [dB]> and **<Min [dB]>** may be used to limit the gain range which can be controlled via the configured DMX Channel.

4.6.3 DMX » Logic (Control Function Block: DMX)

The “DMX » Logic” function block sends 1/TRUE on a logic output whenever an associated DMX channel values reaches or exceeds a configured threshold or 0/FALSE when it falls below that threshold.

The block’s DMX input must be connected with a DMX Source (DMX block, see 0). The active output must be connected to a logic input of the function block to be controlled as shown in the following example:



DMX » Logic parameter dialog:

The parameter dialog allows a channel-by-channel configuration of the following parameters:

Clicking the **[Undo]** button in the Setting window will undo all changes made since the window was opened.

Using the **[∧]** and **[∨]** buttons, you can show or hide the parameters of a channel or channel group in the Settings window.

The **<DMX Channel>** field controls which DMX channel value is to be used for the gain control adaption. DMX Channel accepts values from 1 to 512.

The **<Threshold>** defines which value must be reached or exceeded for the Active output of the block to send TRUE. Thresholds may take values from 0 to 255.

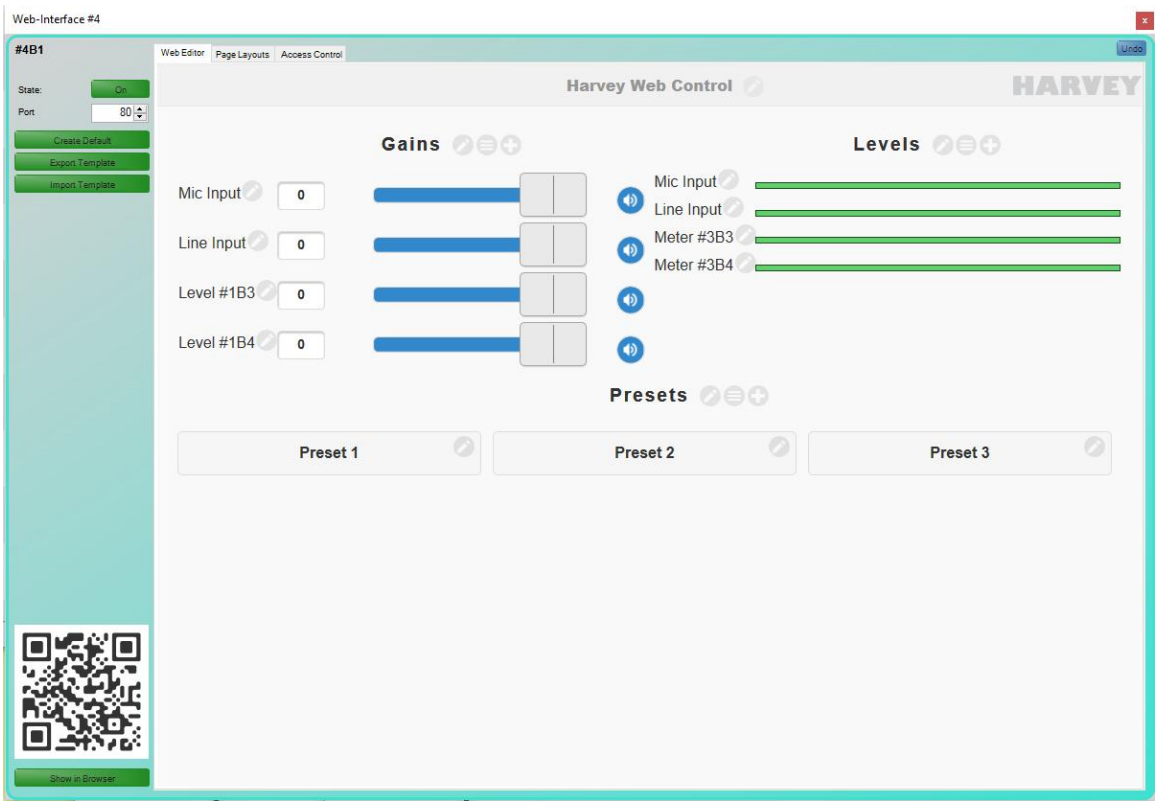
4.7 Control Function Blocks / Controls

Function blocks with control elements are available for controlling the HARVEY devices:

- Web Control
- Remote controls “HARVEY RC4”

4.7.1 Web-Interface (Control Function Blocks / Controls)

HARVEY Composer allows generating and designing a web interface for the current project. That web interface can be used to control the HARVEY project from any web browser on a PC or mobile device.



When the block setting is opened for the first time, the "Page Layouts" tab is open. When you select a layout, the view changes to the "Web Editor" tab.

Web Editor

The Web Editor tab in the settings window provides a what you see is what you get (WYSIWYG) editor to edit your custom web interface. With the help of the button **[Create Default]**, a simple configuration is created from the project. This is displayed graphically in the web editor.

Note: The generated configuration is only updated automatically to a limited extent when the project is changed (see below: *Synchronization with the Project*).

Sections

The web interface contains control elements which are distributed in different sections, delimited by individual headings. The sections are configured with the following symbols next to the headings:

Opens dialog for defining the section heading and switching the arrangement of the controls between horizontally/vertically.

Allows to swap the display position of the section with another section within the selected layout (see below: *Page Layouts*).

Allows to add new control elements to the section.

Control Elements

The control elements in the sections of the Web interface address individual objects within the project. These can be block parameters (e.g. level) or presets. The configuration of individual control elements is done via symbols next to the control elements:

Opens dialog for control element settings. Typically, a label text and the assignment to a project ID (i.e. Block ID or Preset ID) can be defined. Depending on the type of control further settings are possible, e.g. the limitation of the adjustable gain range of a level control.

Hovering the mouse over the pencil icon will show an overlay displaying the following icons:

Removes the control from the section.

Allow to move the control within the section either horizontally or vertically, depending on the section configuration.

Page Layouts

The tab page “Page Layouts” provides a selection of available layout options for the sections within the web interface. The currently selected layout is highlighted with a green border.

Access Control

The access to the web interface may be restricted to a configurable set of devices. Allowed devices can be added to the MAC address list within the tab page “Access Control” either by manually entering the MAC addresses or by scanning the therein provided QR code using the device to be permitted.

If the MAC address list is empty, all devices will be allowed to access the web interface.

Editing the MAC access list may be done online which avoids the need to reload the entire project if single devices are to be added or removed.

Offline Preview

The designed web interface may be previewed using a local web browser or any mobile device by scanning the QR code provided at the bottom-left of the settings window. This allows for testing the web interface on the target device without having a HARVEY mx.16 device at hand.

Synchronization with the project

The contents of the web interface are only updated automatically to a limited extent when the project is changed. Mostly, changes to the project must be transferred manually. This applies in particular to the following aspects:

- ➔ **Important!** Blocks that have been removed in the project *must* be removed manually from the web interface.
  (Otherwise the web interface cannot be updated in online mode)
- ➔ **Important!** Blocks whose IDs have been changed in the project *must* be updated manually in the specific web interface control element settings.
  (Otherwise the web interface cannot be updated in online mode)
- ➔ Block names are not automatically updated in the web interface after changes are made in the project.

➔ Preset names are not automatically updated in the web interface after changes in the project.

The web interface is currently updated automatically for the following controls:

➔ Input Selector: The number of selection buttons and their labeling is updated from the Input Selector Block configuration.

In the case of incorrect references to blocks that are not part of the project, such as a block ID, a note is displayed on the left in the block settings.

Template

Templates can be used to add individual layout, styling and behaviour to the web interface. Without customisation, the current template of HARVEY is always used. Templates can also be used to exchange web interfaces between projects.

With the button **[Export Template]** the complete web interface can be saved in a folder. The current configuration of the web interface including the code for execution and display is saved. The code can be freely changed and adapted and imported back into the project.

With the button **[Import Template]** a complete web interface can be added to the project again. The maximum size of a template is limited to 2MByte, but the template is compressed. If an attempt is made to load a template that is too large, an error message is displayed.

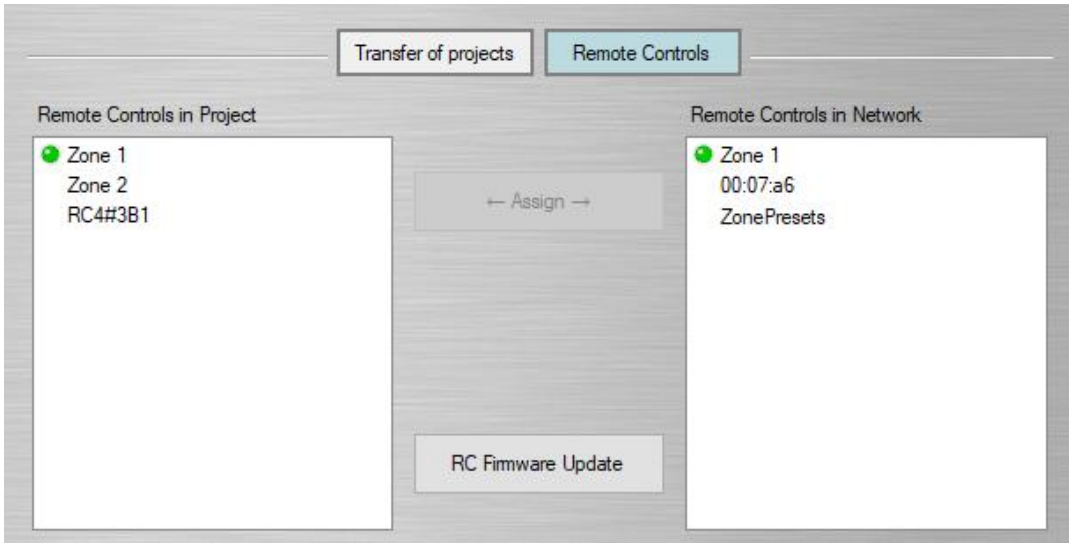
If an older project has been loaded and the block settings are opened for the first time, the template updates automatically. If this is not desired, the **[Undo]** button can be used to undo the template update. With the automatic updating of the template, further developments of the web interface and thus improvements are transferred to an old project and should only be undone if an adapted template has been imported.

4.7.2 RC4/RC12 Remote Control – Introduction

Each HARVEY remote control module (RC4 or RC12) interacts with a HARVEY device over an Ethernet network. Remote control modules can trigger binary actions (on / off) and control audio levels. You can adjust their key colors and the brightness of the RGB LEDs.

To integrate HARVEY remote control modules in a HARVEY project, proceed as follows:

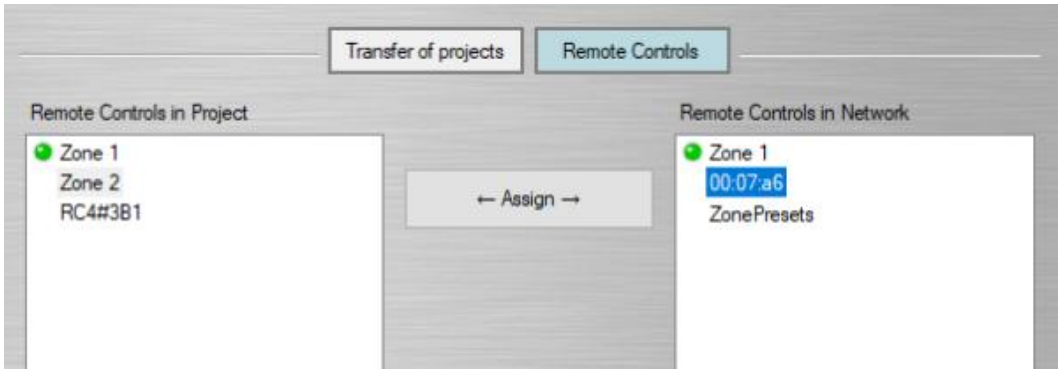
1. Create a HARVEY-Composer project and place a remote-control function block ("RC4", "RC4 DMX", "RC12") on the worksheet for each physical remote control module, parameterize it and connect it to binary control inputs or level controls (see page 100).
2. Connect each physical remote control module to a PoE-enabled 100/1000 Mbps network switch. Connect the network switch to an Ethernet network, which can be reached by the computer running HARVEY Composer.
3. Open the Cockpit and press the button **[Remote Controls]** to call up the area for managing the physical remote control modules whose main purpose is the assignment of project function blocks to physical control modules.



On the left side are all the remote control function blocks that have been placed on the active worksheet of the HARVEY project. In each case, the name given to the function block in the block properties appears (see page 27).

On the right side, all the physical remote controls that HARVEY Composer has found in the network are listed. The designations in the format "xx: yy: zz" indicate the default designation of the remote control - these are the last three values of their MAC address.

4. Click on a function block on the left side and a physical remote control on the right side, so that one label is marked in blue:

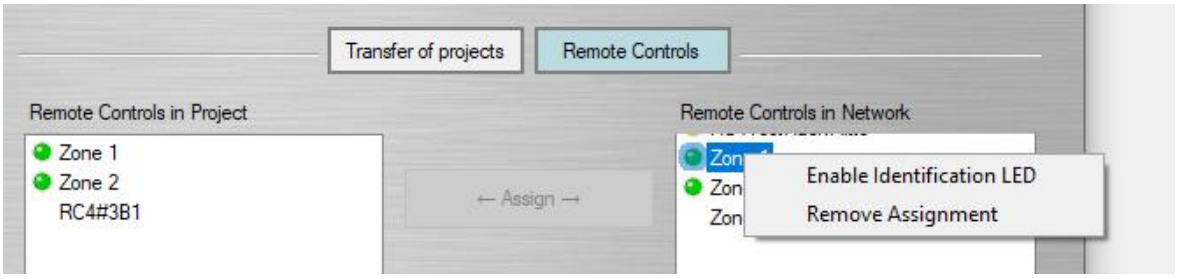


5. Press the **[Assign]** button. Thereafter, the color of the mark changes from none to green, which indicates that both are paired: In addition, the physical remote control takes over the name of the function block.

Note: upon connecting with a HARVEY device the blue status LED on the physical remote control will change to reflect its configured state in the HARVEY project. For that, of course, the HARVEY project has to be running on the device.

6. Transfer the HARVEY project to a HARVEY device (see from page 13).
7. You can now test the function of the remote control in online mode, for example by changing the color of the buttons in the composer parameter dialog or, if necessary, displaying connected level controls in the HARVEY composer and observing the fader movement while turning the endless dial.

In addition to the assignment of operating modules, further actions are possible in the cockpit operating module area:



➔ **Removing the assignment** of RC4 function blocks and physical operating modules: Clicking with the right mouse button on the corresponding module opens a menu selection from which you select "Remove Assignment". After selecting this menu item, the marking of function block and the operating module is removed and the operating module falls back to its default designation.

➔ **Identification of remote controls**: Selecting the menu item "Enable Identification LED" causes the upper light in the light ring around the control dial to illuminate blue so that you can identify the selected module in visual contact.

➔ **Update of the remote control´s firmware**: Selecting **[RC Firmware Update]** starts the update assistant for the remote controls.

Meaning of the color marking:

Next to the names of the RCs you can see one of the following markers:

Marking color	**Meaning**
Green	RC is assigned to the current project in HARVEY Composer
Yellow	RC was previously reached on the network, but has not been sending a response for a short time or is not yet ready to receive configuration requests from HARVEY Composer.
No mark	RC is accessible from HARVEY Composer and is not assigned to the project currently active in Composer*. The RC can be assigned to the project. **Note: The RC may already be assigned to another project which is not currently active in HARVEY Composer.*

4.7.3 RC4 (Control Function Blocks / Controls)

The function block "RC4" represents a PoE-powered, network-based HARVEY RC4 remote control.

The function block transfers the state of the hardware-related key states to the software level. A pressed key is interpreted as "1 / TRUE" and a key that is not pressed is interpreted as "0 / FALSE". A pressed button ("Button1" ... "Button4") is interpreted as "1 / TRUE" and a button that is not pressed is interpreted as "0 / FALSE". The push-button of the rotary knob ("Mute") is intended for the mute of level controls. It signals its state by toggled switching: after pressing and releasing, it signals "1 / TRUE" and after the next pressure-release sequence it signals "0 / FALSE". The state of the endless rotary encoder can be connected to the level control inputs of level controls. Any number of RC4 function blocks can be used in a HARVEY Composer worksheet, with a maximum of 64 RC4s being grouped into one block. RC4 can be grouped arbitrarily, which creates a synchronous control and status display on the RC4 grouped into groups.

System Integration

The physical control modules are integrated into the project in the cockpit under "Remote Controls". See section 4.7.2.

➔ Tip: Make use of entering a name for the RC4 function block (see "Block properties" on page 27). The name entered there appears in the control module area of the Cockpit. If no designation is given, the generic identifier appears there, such as "RC4#4B1".

RC4 parameter dialog

[Undo] restores the full parameter state that was in effect before the window was opened.

[^] / **[v]** shows or hides the parameter overview of a channel or group in the settings.

[Level Control Min dB] sets the minimum level control level to be reached when the endless encoder turns to the left.

[Level Control Max dB] sets the maximum level control level to be reached when the endless encoder turns clockwise.

If you press one of the 4 buttons of the **[Button Color]**, a dialog appears for selecting a color from a color palette for the respective button. The button on the far left corresponds to the upper button of the RC4 remote control.

If Button Color is set to **Switching** mode, an **Active** color may additionally be configured which will be displayed on the RC4 when the attached control wire receives a status 1/TRUE from the attached block.

In Online mode the current RC coloring will be displayed under **Current**.

In the **[Brightness]** box, the brightness of the RC4 remote control RGB LED is set in the range of 0% to 100%.

➔ The Function block "RC4" provides connections in the control layer.

Name	**Control Port**	**Range**	**Status (1)**
Mute	Logic Output *not exlusive (2)*	On/Off, toggling	Display mute state
Level	Level Output *not exclusive (2)*	Defined by Min/Max	Display level gain
Button 1 to 4	Logic Output	On/Off	Triggers color switching if Button Color is configured as Switching

(1) The control ports can receive state/status feedback from the connected sinks and visualize it on the RC4.

(2) Connected mute or level parameters are not exclusively controlled by the RC4 block and can still be parameterized via other ways. (e.g. H-Text or presets).

4.7.4 RC4-DMX (Control Function Blocks / Controls)

The function block "RC4-DMX" represents a PoE-powered, network-based HARVEY RC4 remote control for DMX light control applications.

The function block transfers the state of the hardware-related key states to the software level. A pressed key is interpreted as "1 / TRUE" and a key that is not pressed is interpreted as "0 / FALSE". A pressed button ("Button1" ... "Button4") is interpreted as "1 / TRUE" and a button that is not pressed is interpreted as "0 / FALSE". The push-button of the rotary knob is intended for enabling / disabling the light by calling adequately configured presets. It signals its state by toggled switching: after pressing and releasing, it signals "1 / TRUE" and after the next pressure-release sequence it signals "0 / FALSE". The state of the endless rotary encoder can be connected to any channel input of a DMX block. Any number of RC4-DMX function blocks can be used in a HARVEY Composer worksheet, with a maximum of 64 RC4-DMX blocks being grouped into one block. RC4-DMX can be grouped arbitrarily, which creates a synchronous control and status display on the RC4 grouped into groups.

System Integration

The physical control modules are integrated into the project in the cockpit under "Remote Controls". See section 4.7.2.

➔ Tip: Make use of entering a name for the RC4-DMX function block (see "Block properties" on page 27). The name entered there appears in the control module area of the Cockpit. If no designation is given, the generic identifier appears there, such as "RC4#4B1".

RC4-DMX parameter dialog

[Undo] restores the full parameter state that was in effect before the window was opened.

[^] / **[v]** shows or hides the parameter overview of a channel or group in the settings.

[DMX Control Min] sets the minimum DMX value to be reached when the endless encoder turns to the left.

[DMX Control Max] sets the maximum DMX value to be reached when the endless encoder turns clockwise.

If you press one of the 4 buttons of the **[Button Color]**, a dialog appears for selecting a color from a color palette for the respective button. The button on the far left corresponds to the upper button of the physical RC4 remote control.

If Button Color is set to **Switching** mode, an **Active** color may additionally be configured which will be displayed on the RC4 when the attached control wire receives a status 1/TRUE from the attached block.

In Online mode the current RC coloring will be displayed under **Current**.

In the **[Brightness]** box, the brightness of the RC4 remote control RGB LED is set in the range of 0% to 100%.

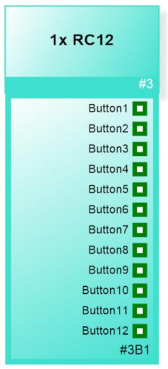
➔ The Function block "RC4" provides connections in the control layer.

Name	**Control Port**	**Range**	**Status (1)**
Enable	Logic Output *not exlusive (2)*	On/Off, toggling	Display On/Off state
DMX	DMX value Output *not exclusive (2)*	Defined by Min/Max	Display DMX value
Button 1 to 4	Logic Output	On/Off	Triggers color switching if Button Color is configured as Switching

(1) The control ports can receive state/status feedback from the connected sinks and visualize it on the RC4.

(2) Connected enable or dmx parameters are not exclusively controlled by the RC4 block and can still be parameterized via other ways. (e.g. H-Text or presets).

4.7.5 RC12 (Control Function Blocks / Controls)



The function block "RC12" represents a PoE-powered, network-based HARVEY RC12 remote control.

The function block transfers the state of the hardware-related key states to the software level. A pressed key is interpreted as "1 / TRUE" and a key that is not pressed is interpreted as "0 / FALSE". A pressed button ("Button1" ... "Button12") is interpreted as "1 / TRUE" and a button that is not pressed is interpreted as "0 / FALSE".

Any number of RC12 function blocks can be used in a HARVEY Composer worksheet. Multiple RC12s can be grouped into one block virtually as channels. As known from other blocks, RC12 channels can be grouped arbitrarily, which creates a synchronous control and status display on the RC12 modules being grouped.

System Integration

The physical control modules are integrated into the project in the cockpit under "Remote Controls". See section 4.7.2.

➔ Tip: Make use of entering a name for the RC12 function block (see "Block properties" on page 27). The name entered there appears in the control module area of the Cockpit. If no name is given, the generic identifier appears there, such as "RC12#12B1".

Names for multiple RC12 modules within a block (“channels”) may be delimited with a comma in the Description field.

RC12 parameter dialog

[Undo] restores the full parameter state that was in effect before the window was opened.

[^] / **[v]** shows or hides the parameter overview of a channel or group in the settings.

If you press one of the 12 buttons of the **[Button Color]**, a dialog appears for selecting a color from a color palette for the respective button. The button layout matches that on the remote. The numbering corresponds to the numbers shown on the worksheet.

If Button Color is set to **Switching** mode, an **Active** color may additionally be configured which will be displayed on the RC12 when the attached control wire receives a status 1/TRUE from the attached block.

In Online mode the current RC12 coloring will be displayed under **Current**.

In the **[Brightness]** box, the brightness of the RC12 remote control RGB LED is set in the range of 0% to 100%.

➔ The Function block "RC12" provides connections in the control layer.

Parameter	Control Port	Range	Status (1)
Button 1 to 12	Logic Output	On/Off	Triggers color switching if Button Color is configured as Switching

(1) Die Taster-Anschlüsse können Zustand/Status-Rückmeldungen von den angeschlossenen Senken empfangen und auf der RC12 als umschaltende Tastenfarben visualisieren, wenn der Block entsprechend parametriert ist.

5 Controlling and Integrating Media Systems

This section describes exemplary control scenarios in order to explain the integration of media systems. Be sure to read the following paragraphs to get an idea of the impressing control and conversion features your HARVEY system offers.

5.1 Interface-Format Conversion

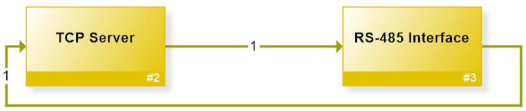
The data exchange between your HARVEY device and other devices is straightforward. The various serial interface options of the HARVEY device allows for networking devices for communication and control purposes. This is particularly true for the Ethernet interface used in combination with the TCP protocol.

➔ You can control several HARVEY devices and third-party devices connected to their serial interface ports using LAN and Wi-Fi links.

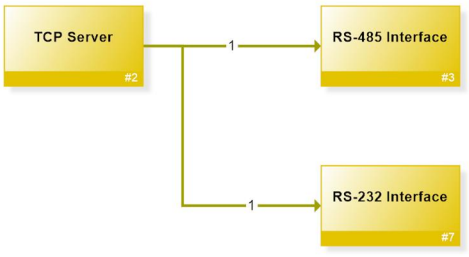
Interface-format conversion is very simple to implement using your HARVEY system.

➔ Simply connect the output and input ports of different interfaces

In this example data will be forwarded from an incoming TCP stream to the RS-485 interface. To achieve bidirectional data conversion you have to draw wires in both directions as shown in the following example:



You may also connect multiple interfaces, e.g. to broadcast signals from one interface to all others:



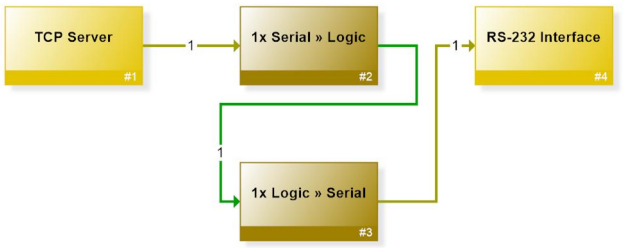
5.2 Integrating Proprietary Protocols

In this scenario, you want to operate a DVD player equipped with an RS-232 port from an iPad through a Wi-Fi link. Provided the Wi-Fi is on the same network, you can use the HARVEY mx.16 for receiving, translating, and forwarding control messages.

iPad sends DVD player understands ***08***

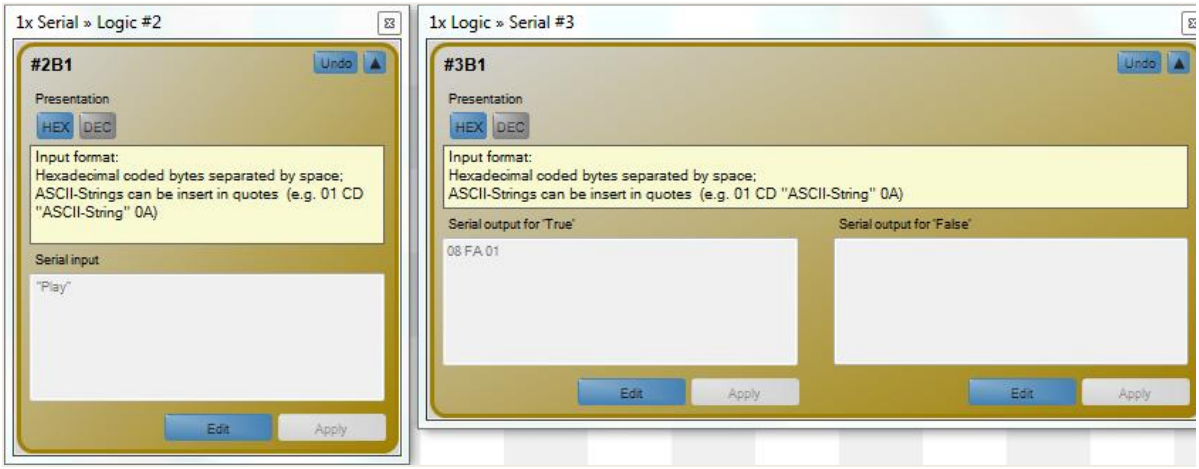
Using the interface-format conversion functionality of the HARVEY mx.16, converting Ethernet information into the RS-232 format is straightforward.

The problem is that the DVD player expects a play message that is different from the one transmitted by the iPad. However, you can configure a translation function using the Serial » Logic and Logic » Serial function blocks:



Insert the Serial » Logic and Logic » Serial blocks between the Ethernet and RS-232 blocks.

The condition entered into the Serial Input field of the Serial » Logic block is the ASCII string “Play”. This corresponds to the expected message sent by the iPad. If the condition is met, i.e. the “Play” string is received, the block changes the logical output to 1/TRUE. In doing so, it triggers the transmission of the data packet that has been defined for that status on the downstream Logic » Serial function block: On that block, a data package containing the “08 FA 01” string (which corresponds to the play message for the DVD player) was entered into the Serial Output for 'True' field. Next, the triggered data packet is transmitted to the RS-232 interface and from there sent to the DVD player. The player receives a play message in a format it understands and consequently starts playing.



See page 115 for details on the Serial » Logic and Logic » Serial function blocks.

➔ Tip: Using specialized apps such as iCue by Cue a.s., you can create your own touch-enabled GUIs with objects and thus transmit user-defined control messages through TCP. This way, you can configure attractive user interfaces for applications of your HARVEY installation.

6 HARVEY Device Variants

The following table lists the differences between the HARVEY device variants and their behavioral impacts on the HARVEY Composer.

Property	**HARVEY Pro NxM [Dante][Amp][AES]**	**HARVEY mx.16[Dante][Cobranet]**	**Impact on Composer-Block**
Number of Analog Audio Channels	N analog inputs in steps of 4 M analog outputs in steps of 8 Up to a sum of 32 channels (N+M = 32)	16 analog inputs 16 analog outputs	Number of inputs and outputs (sections 3.1.1 and section 3.1.2)
Dante Option	64 input channels 64 output channels	16 input channels 16 output channels	Availablity of Dante-Block (section 3.1.5)
AES3 Option	2 input channels 2 output channels	Not available	Availability of AES3 blocks (section 3.1.6)
CobraNet Option	Not available	16 input channels 16 output channels	Availability of Cobranet-Block (sections 3.1.3 and section 3.1.4)
Mic Inputs (Number und Gain)	Each analog input is mic capable (P48 and gain) Gain range: 0..60 dB in 3 dB steps	Only the first eight channels are mic capable (P48 and gain) Gain range: 0, 10..65 dB in 1 dB steps	Parameter input block (section 3.1.1)
Line Inputs (Gain-Steps)	Each analog input is line level capable. Gain steps: 0, 6, 9, 15 dB	Each analog input is line level capable. Gain steps: 0, 9, 15, 18 dB	Parameter input block (section 3.1.1)
Line Outputs (Gain Steps)	Each analog output is line level capable. Gain steps: 0, -6, -9, -15 dB	Each analog output is line level capable. Gain steps: 0, -9, -15, -18 dB	Parameter input block (section 3.1.2)
Switching Inputs	Four switching inputs	Eight switching inputs	Number of switching inputs (section 4.1.1)
Switching Outputs	Six switching outputs, of which - Four transistor outputs, each with two connection options: supplied internally, externally powered - two relay switches	Three switching outputs - no transistor outputs - three relay switches	Type and number of switching outputs (section 4.1.2)
Voltage Inputs	Two voltage inputs	Eight voltage inputs	Number of voltage inputs (section 4.1.3)