# APPolo | Control™

# Auto **Control**
# GPI **Control** User Guide

| | |
|---|---|
| Revision: | 1.0 RELEASE |
| Last Updated: | February 2014 |
| Service Contact: | support@lynx-technik.com |
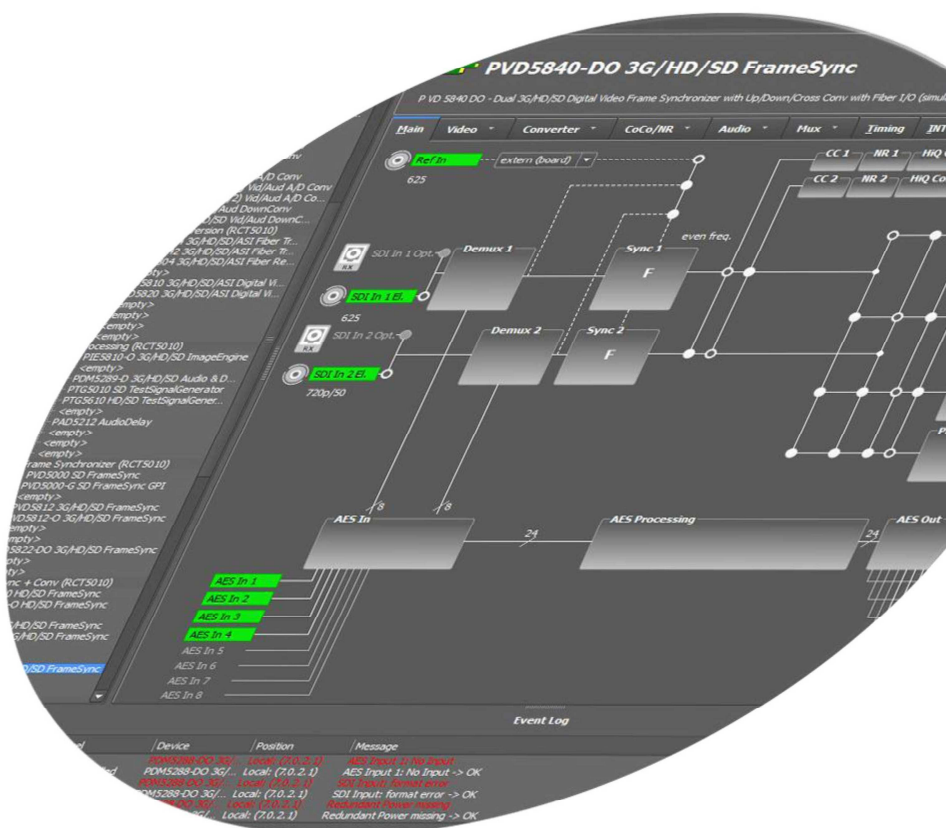
## Contents

## Introduction

AutoControl is part of the LYNX APPolo ControlSystem. General information on the LYNX APPolo ControlSystem can be found on http://appolo.lynx-technik.com.

AutoControl for APPolo is a radical advancement for infrastructure control and automation. Most facilities use automation yet generally do not have the ability to automate static terminal equipment. AutoControl adds automation and programmable intelligence inside the terminal equipment, which enables automatic reconfiguration of the modules functions, signal routing and internal signal processing. This opens up an entirely new layer of power and flexibility to automated facility control.

APPolo monitors every module's internal status and configuration, every input and output in terms of presence and video format, plus multiple external GPI inputs in the system. Based on these input parameters, the user can program an automatic conditional "action" using logical functions.

A simple example: The video input to a module has embedded audio present, however the embedded audio stream 1 could be dropped or go missing in the input signal. The user would like to automatically embed an external audio input if the SDI input is missing this embedded audio.

With APPolo AutoControl a simple solution is available. APPolo can be programmed to monitor the embedded input stream 1 on the selected video input. If this audio is present, the system will do nothing and transparently pass the audio to the output. But if this audio is missing, then the system will automatically reconfigure the

module and route an external audio input to the embedder and embed this alternative audio instead.

There is virtually no limit to the automation capabilities that APPolo offers! Everything is possible, from a simple action (as in the above example) to a very complex series of events that could also span multiple modules across multiple rack frames in several locations.

## Getting Started

AutoControl and GPIControl are both functionalities of the LYNX SERVER. The LYNX SERVER can be added to any RCT5023 RackController as an optional hardware-extension board (OH_RCT5023_SERVER). The SERVER is already integrated in the the RCT5031 MasterController Card. A SERVER can control all modules in all of the RackFrames that are logically connected to this SERVER. Please refer to the "LYNX APPolo Infrastructure Guide" or visit http://appolo.lynx-technik.com for more information.

GPIControl (explained in the next section) is the simplest variant of AutoControl and works based on GPI-Inputs. GPI-Inputs are provided from the LYNX RackFrame through the RCT5023-G controller board. Consequently, every RackFrame which is equipped with a RCT5023-G RackController provides up to 9 GPI-Inputs to the AutoControl infrastructure.

*NOTE: GPI is used here in the sense of "General Purpose Interface". Accordingly, there are GPI-Inputs (external triggers are sensed by the LYNX System) and GPI-Outputs (internal triggers can be used to control external equipment).*

The regular AutoControl provides even more flexibility than GPIControl. See section "AutoControl Configuration" on page 9 for more details on AutoControl.

### GPIControl

The RCT5023-G RackController provides 9 GPI Input connections on each LYNX RFR5012 RackFrame. Each such GPI Input contact can be configured to trigger one or multiple actions within the system. As an example:  One GPI Input can be configured to switch the audio embedder in a specific card to ON. Based on this AutoControl RULE, the embedder will be enabled whenever that GPI Input is triggered.

## GPI Connections (physical)

This paragraph explains the physical GPI Input and Output connections.

The connector panel of the RFR5012 RackFrame has a total of five RJ45 connectors. One of them (marked in yellow color) is the Ethernet/LAN connection. When a RCT5023-G RackController Card is used into the RackFrame, the two outer RJ45 connectors in the row of four (indicated by the red circles in figure 1) are the GPI connections.



figure 1: RackFrame connections

*NOTE:* *The description on the back of the Rack-Frames may look different from the picture shown here. I.e. the four RJ45 connectors might be labelled as "Control" and "Extend", using green and blue colors. These descriptions are used for older Controller Cards (e.g. RCT5021, RCT5031 etc.). GPI contacts are not available with these older Controller Cards. When using the current RCT5023-G RackController, then the above picture shows the correct usage of these RJ45 ports (even if the label still shows the older descriptions). Please contact LYNX Technik support to order a replacement sticker for your RackFrame.*

*NOTE:* *When a RCT5023 RackController (i.e. not the RCT5023-G Version) is inserted into the RackFrame, then these four RJ45 connectors are without any functionality.*
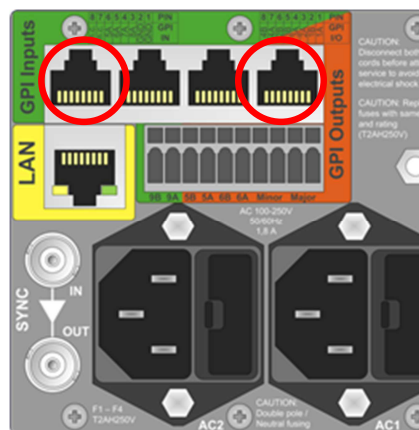
### GPI Inputs

Each RackFrame with a RCT5023-G RackController provides 9 independent GPI inputs. In order to activate any of the GPI inputs, an electrical closure between the common ground ('GND') and the individual GPI contact 'A' has to be provided. The next section explains how APPolo GPIControl can be used to program one or more actions that shall be activated when an individual GPI input is activated.

The first 4 GPI Inputs (i.e. GPI #1, #2, #3, #4) can optionally be configured to be GPI Outputs instead of GPI Inputs. See below.

### GPI Outputs

Each RackFrame with a RCT5023-G RackController provides up to 6 programmable GPI Outputs. The next section explains how APPolo GPIControl can be programmed to activate an individual GPI Output under pre-defined conditions. When a GPI Output is activated, the appropriate 'A' and 'B' contacts are connected (by relais-closure) inside the RackFrame. These contacts are electrically isolated from the

internal electronics of the RackFrame. The first 4 GPI Outputs are actually configured as GPI Inputs by default. Use APPolo GUI to configure any one of these contacts to be a GPI Output instead of a GPI Input.

## GPI Control Configuration

In the APPolo GUI, click onto the icon  in the toolbar or select "Auto Control" from the "Tools" menu. You will be redirected to the setup page for "GPI Control" and "Auto Control".



figure 2: one Tab per Server

This page displays one tab per connected Server (circled in figure 2). Each individual Server can be identified by its IP address.

Now place the mouse over the tab of a Server. This will bring up a list of GPI-enabled RackFrames that are attached to this Server (figure 3). After selecting one of the GPI controller cards from this list, the column at the far left will show a list of 9 configurable GPIs for this RackFrame (marked in figure 5).



figure 3: list of connected GPI Frames

To start editing, press the button that is labelled "Edit Configuration …" just above this list of GPIs (figure 4). You will have to enter the administrator password to continue.
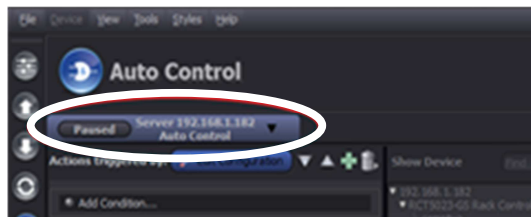


figure 4: enable EDIT mode

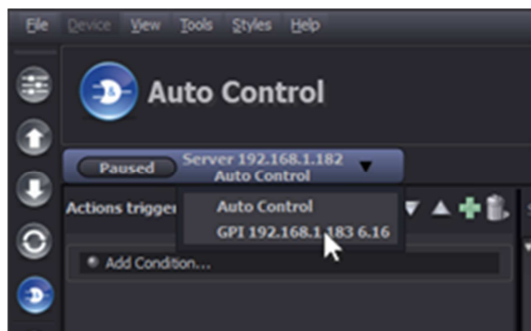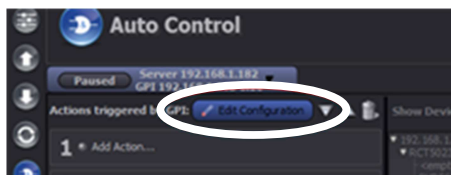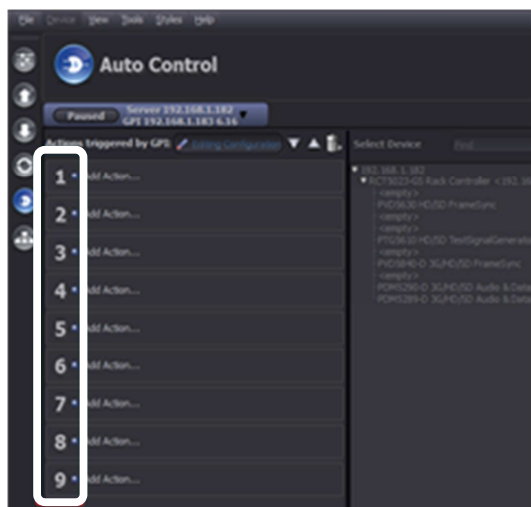

figure 5: GPI Inputs per RackFrame

To start configuring the actions that are to be triggered by GPI Input 1, click on "Add Action…" positioned to the right of the "1" indicating GPI 1. This will activate the device tree to the right of the GPI list (marked in figure 6).

All RackFrames and all Devices that are connected to the Server are shown here and can be selected (i.e. regardless of the type of RackController)
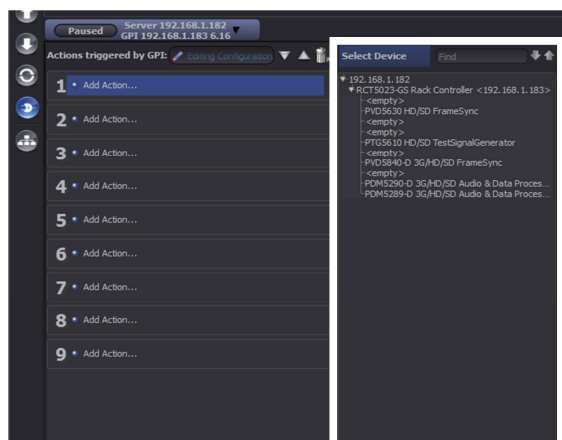
figure 6: specify Device for Action

As soon as a device has been selected from the device tree, the column to the right will display all the parameters of this module that can be modified by this Action (i.e. READ-ONLY parameters are not shown here).

Depending on the complexity of this module, the list of parameters can be quite long. A filter function is provided to help find the required parameter (figure 7). Press "Clear Filter" to show the complete list again.
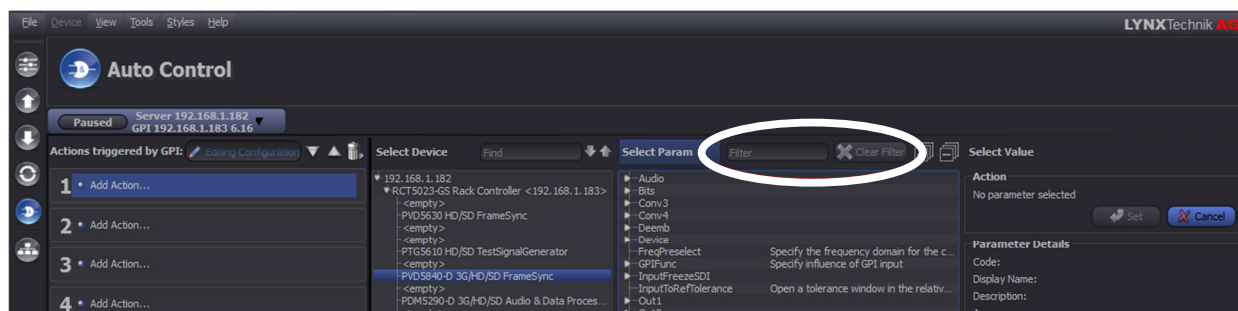
figure 7: use Text-Filter to reduce list of parameters

Once a parameter has been selected, the furthest right side of the GUI allows you to select a specific value (see figure 8).

*Note    Certain parameters can be in a "read-only" state depending on other circumstances. (e.g. the "audio bit depth" selection of an embedder [20/24bit] can only be modified for SDTV video. For HDTV video it is fixed to 24Bit). Such parameters can still be selected for GPI Control. However, if they are in a "read-only" state when the GPI control is triggered, they will simply not be modified.*
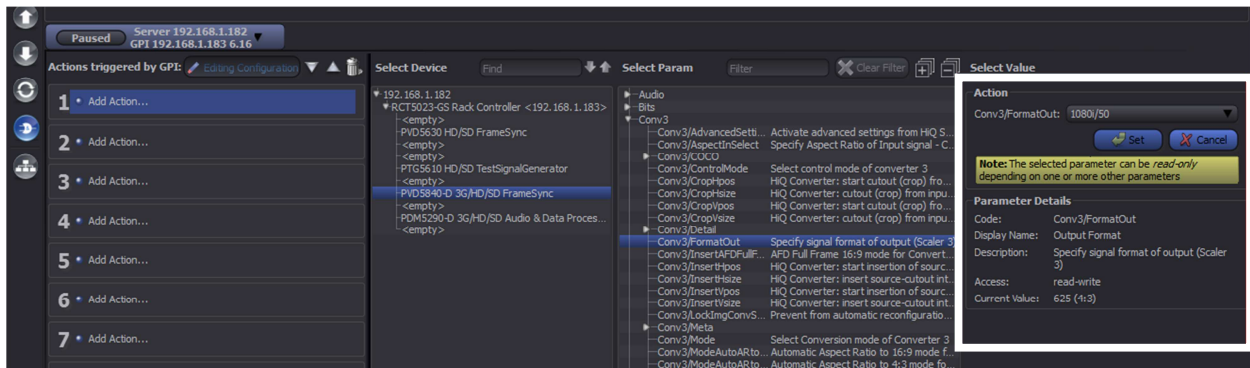
figure 8: specify target value for GPI Input to set

Finish your definition by clicking the SET button (figure 8). The selected parameter of the selected device will now be set (enforced) to this specified value while the GPI Input is ACTIVE. An appropriate short description is displayed next to the GPI number for which it has been configured (figure 9).
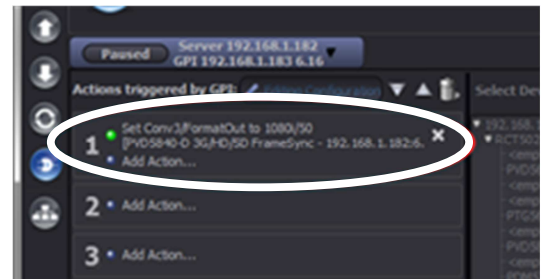


figure 9: description of individual Action

To review the detailed definition of an Action, simply click onto this description text (figure 9) and the device tree, parameter list and value selection will be shown again, with a highlight on the selected device, parameter and value.

Any number of Actions can be associated with each GPI Input. Simply click on "Add Action …" again to add another Action next to an existing one. All associated Actions will be executed (enforced) simultaneously, while the GPI Input is Active.
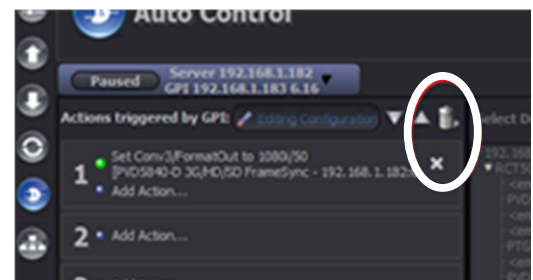


figure 10: delete individual Action

An individual Action can be removed from a GPI Input by clicking the "X" symbol next to it, or by dragging it onto the waste-bin symbol above (see figure 10).

*Note*   *The AutoControl Engine (process in the Server) has to be ACTIVATED to enable this automatic process. See next section for detals.*

## AutoControl Engine: general de-/activation

Each LYNX SERVER runs a process called the "AutoControl Engine". This process is responsible for monitoring GPIs and other programmed conditions. And this process will execute the specified Actions according to the current AutoControl / GPIControl specifications.

The AutoControl Engine of a SERVER can be activated or de-activated by the Administrator. As long as the AutoControl Engine is in de-activated mode, no programmed Actions will be executed.

The AutoControl Engine needs to be ACTIVE in order to do its pre-programmed work.

To activate the AutoControl Engine, use the button to the left of the IP address of the



figure 11: activating the AutoControl Engine

Server (figure 11). The AutoControl Engine is active when this button is green and shows the word "Active". It is deactivated when the button is gray and shows the word "Paused".
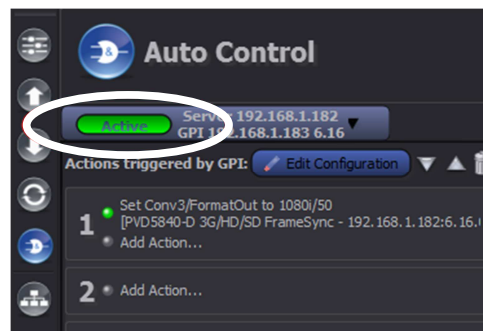
## AutoControl vs. GPI Control

AutoControl works on the same principle as the above GPI Control: I.e. an event triggers one or more pre-programmed Actions. The only difference between GPI Control and AutoControl is that in the case of AutoControl the trigger is not derived from a GPI Input. Instead, the trigger to start the pre-defined Action(s) is derived from the current status of any parameter within the APPolo Control System.

AutoControl requires a LYNX SERVER running APPolo 7.0.0 or newer (just like GPI Control). In addition, AutoControl requires the Option OC_SERVER_AC_BASE to be enabled in that SERVER (while GPI Control does not require any Option-Code).

All LYNX Cards in all RackFrames that are connected to this SERVER can be used for AutoControl (both to supply conditional triggers as well as to be controlled via pre-programmed Actions).

## AutoControl Configuration

The configuration of AutoControl is done in the same Auto Control Editor shown above. (i.e. select "Auto Control" from the "Tools" menu).

To show the AutoControl Editor GUI, move the mouse pointer over the tab with the IP address of the designated server and select "AutoControl".
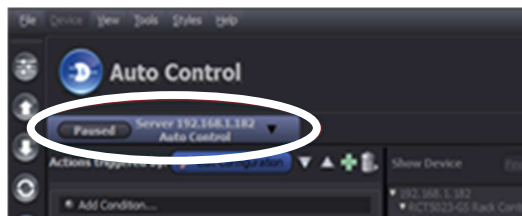


figure 12: AutoControl Editor

On first sight the editor for AutoControl looks very similar to the GPI Control editor. However, instead of having a fixed list of 9 GPI Inputs as predefined trigger-conditions, there's only one entry here, labelled "Add Condition… ".

First, click the "Edit Configuration" button and supply the "admin" password, in order to start modifying the AutoControl configuration.

Now you can define the first Rule by defining a Condition that will be the trigger. The workflow is similar the same as described above for defining an Action:

1) Click "Add Condition…" and the device tree to the right will be activated.

2) Select the device from which you want to pick your trigger. Now the parameter list of this device will be show in the column to the right *(including all read-only parameters, which have not been visible when specifying the Action, see described above).*

3) Select the desired parameter from this list, and the parameter details and selectable values are available on the far right of the GUI.

4) Specify the value of this parameter that shall trigger the Condition to become TRUE.

5) To finalize the configuration of the Condition, click the "Select" button on the far right.

At this point, a first Condition has been defined. This means that as soon as the current value of the selected parameter (in the named device) equals the specified value -> the Condition will evaluate to TRUE. And this status will then be able enforce one or more Actions (just like an active GPI Input can do, see above). Such Actions have yet to be programmed though. This is done in the exact same way as described in the context of GPI Control (above):

1) Click "Add Action …"
2) Select Device
3) Select (writable) Parameter inside Device
4) Specify target value for Parameter.
5) Click "Set"

*NOTE: Section "Example Configuration (Simulation)" on page 11 shows an easy example.*

## Priority of Execution

If more than one Rule is defined in AutoControl, it is possible that multiple Conditions evaluate to TRUE at the same time, enforcing multiple Actions at the same time. And it is possible that these (independent) definitions of Actions actually contradict each other. E.g. Rule #1 could enforce parameter "X" to value "1", while Rule #2 wants to set the same parameter to a different value.

In this case, the Rule that appears first in the list (top to bottom) takes precedence over other Rules that appear further down in the list.

Use the UP / DOWN buttons to re-arrange the order in which the Rules appear in the list. This will re-arrange the order of priorities between these Rules accordingly.

## Example Configuration (Simulation)

This section provides instructions on how to setup a simple AutoControl scenario, based on simulated devices in the APPolo GUI. No LYNX hardware (RackFrame or Controller) is required for this simulated example. Going through these instructions step-by-step will provide an easy quick-start to the capabilities of AutoControl.

In this example, a testsignal generator in one CardModule (type CAD5320) is turned ON / OFF, depending on the current signal input presence status in a second CardModule (type DAA5320). While this is just an arbitrary example, it shows the practical use of AutoControl.

Step 1 (figure 13)

In APPolo GUI, enable the simulation of Devices (from the "View" menu).

This will produce a list of simulated devices in the Main-tree. These simulated devices are provided here for demonstration purposes.
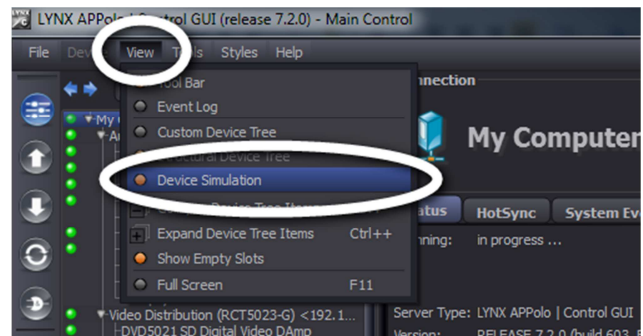


figure 13: Example, Step 1: enable Simulation

Step 2 (figure 14):

Start AutoControl Editor from Menu "Tools"



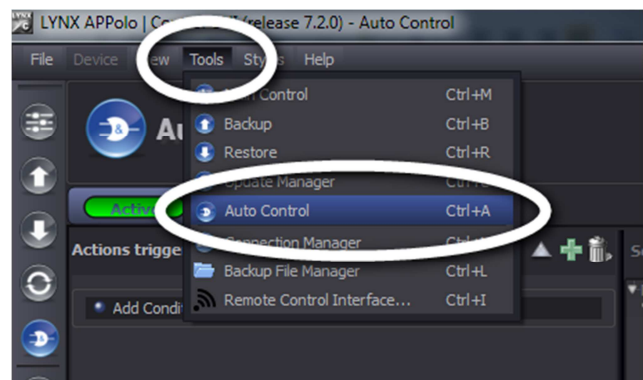figure 14: Example, Step 2: start AutoControl Editor
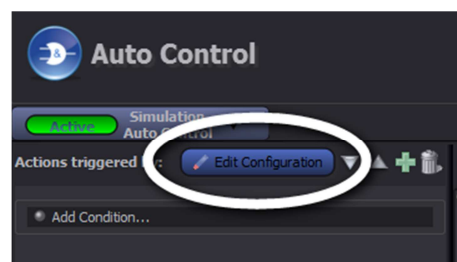
Step 3 (figure 15):

Enable AutoControl Editing mode



figure 15: Example, Step 3: Start Editing

## Step 4 (figure 16)

Specify 1st Condition:

A    click "New Condition …"
B    select device DAA5320
C    select Parameter "In1"
D    select Value "missing"
E    click "Select"
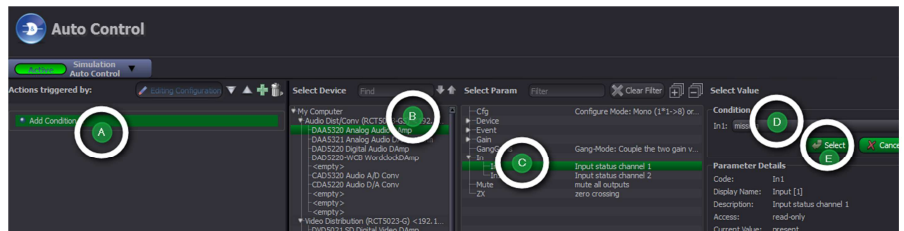


figure 16: Example, Step 4: specify first Condition

## Step 5 (figure 17)

Specify 1st Action:

A    click "Add Action …"
B    select device CAD5320
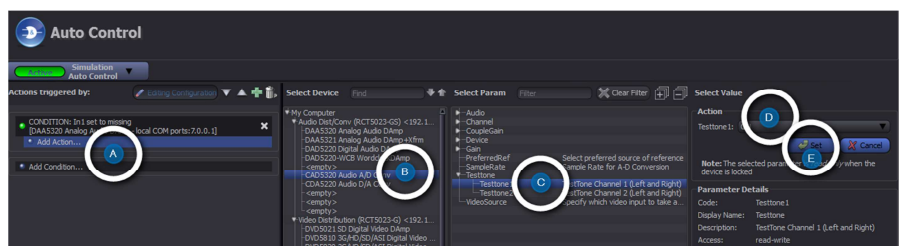C    select Param. "Testtone1"
D    select Value "on"
E    click "Set"



figure 17: Example, Step 5: specify Action for first Condition

## Step 6 (figure 18)

Specify 2nd Condition:

A    click "Add Action …"
B    select DAA5320
C    select param "In1"
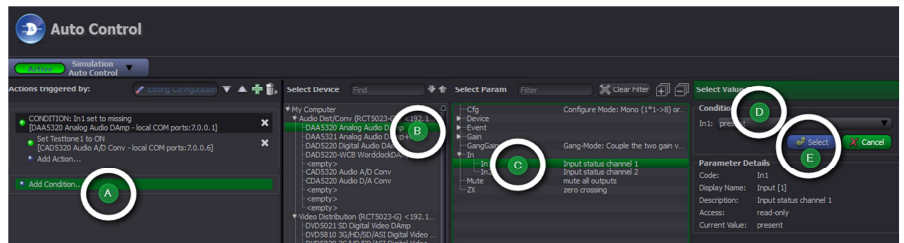D    select Value "present"
E    click "Set"



figure 18: Example, Step 6: specify second Condition

## Step 7 (figure 19)

Specify 2nd Action:

A    click "Add Action …"
B    select CAD5320
C    select param "Testtone1"
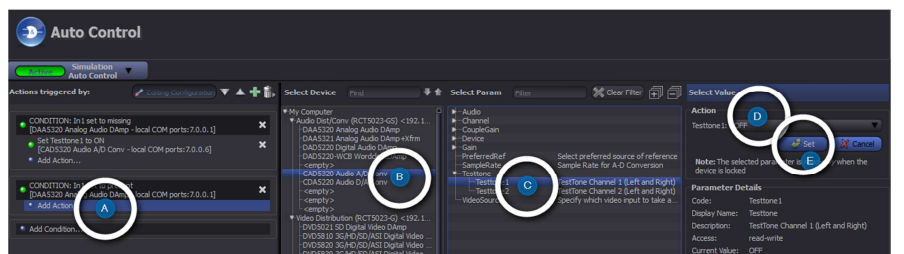D    select Value "off"
E    click "Set"



figure 19: Example, Step 7: specify Action for second Condition

Step 8 (figure 20)

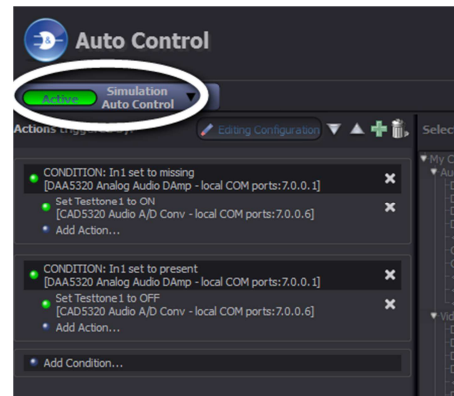Make sure that the AutoControl Engine is set to ACTIVE.



figure 20: Example, Step 8:
set AutoControl Engine
to ACTIVE

Now the AutoControl configuration is complete and active. Notice that the Conditions and Actions are listed here, so that you can read what they do:

1) The first Rule triggers on the Condition that the input signal (Parameter "In1") of the DAA5320 has the state "missing". If this is the case, the only defined Action will set the Testsignal Generator (Parameter "Testtone1") of the CAD5320 to "ON".

2) The second Rule has been programmed to do the opposite: whenever the input signal of the DAA5320 is "present", the Testsignal Generator of the CAD5330 will be turned "OFF".

The following steps set up a little demonstration of this automatic behavior. Here we make use of the fact, that these are "simulated" devices (not real CardModules). With such simulated devices, it is possible to simulate the "Input missing" Event.

Step 9 (figure 21)

Open an extra control window for the DAA5320

- in the MAIN-View (Tools menu)
- click the Right-Mouse-Button onto the DAA5320
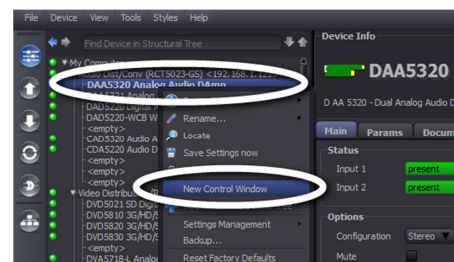- select "New Control Window"



figure 21: Example, Step 9:Open
extra window for DAA5320

Step 10 (figure 21)

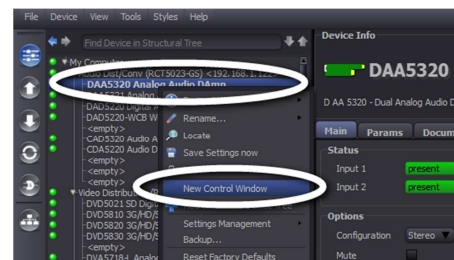Open an extra control window for the CAD5320 in the same way.



figure 22: Example, Step 10:Open
extra window for CAD5320

Step 11 (figure 23)

- Arrange the two Extra-Windows to the right side of the APPolo Main-Window as shown in figure 23.

- In the APPolo Main-Window, select the DAA5320 from the tree and open the "Events" tab.

- Simulate a missing input by clicking the "Simulate Event" checkbox.

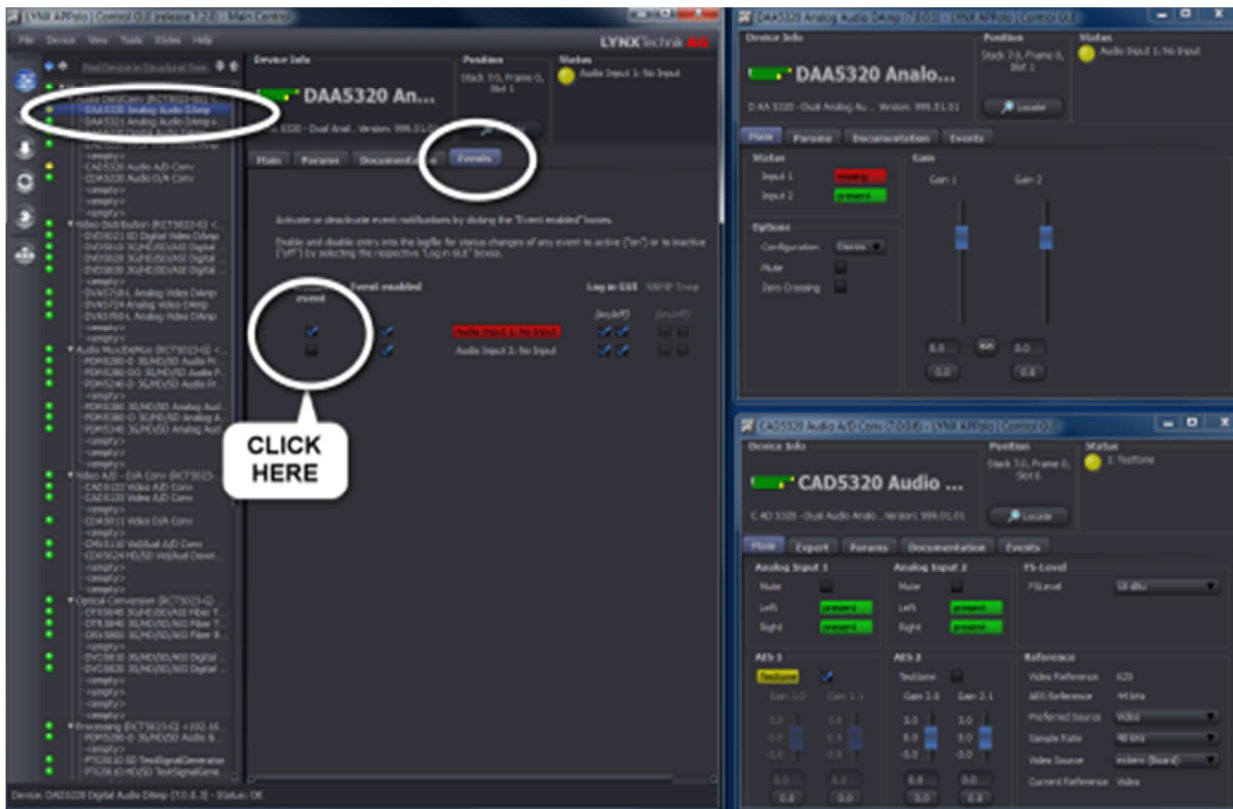➔ Watch how the AutoControl Actions are enforced in the CAD5320.



figure 23:  Example, Step 11:Arrange windows -> see AutoControl in Action

| | |
|---|---|
| *Document* | *LYNX_APPolo_UserGuide_AutoControl.docx* |
| *Last Printed:* | *13.02.2014 09:11* |