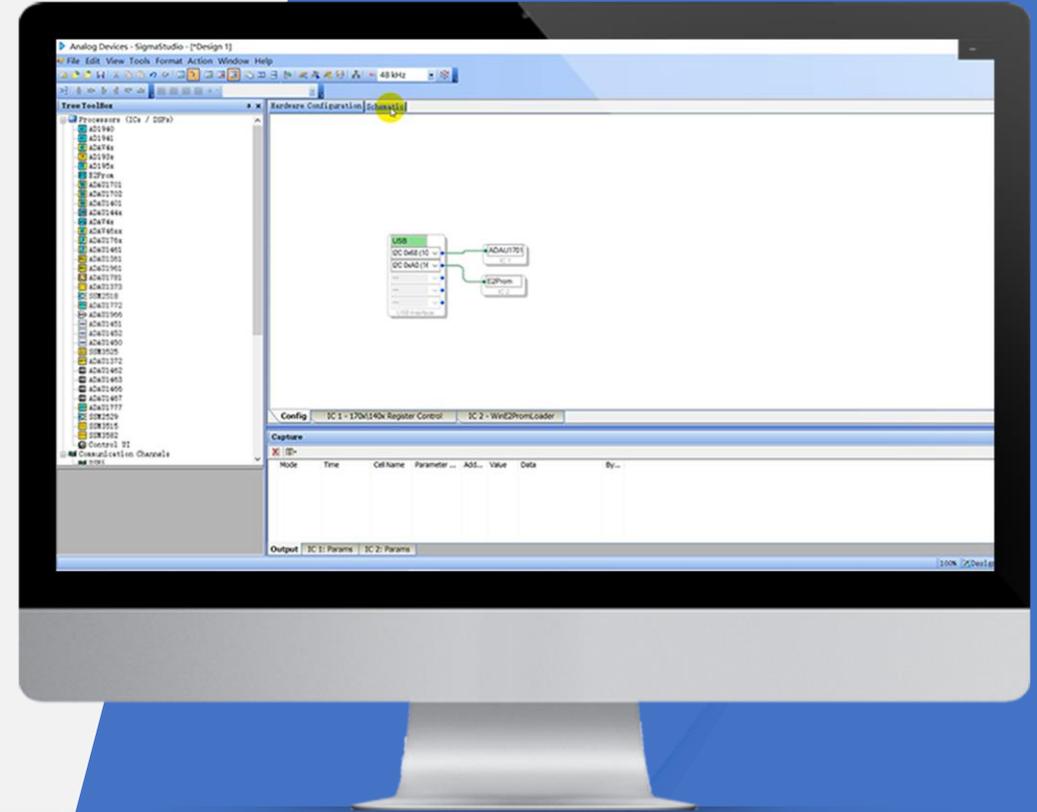


# How to Program WONDOM Products with ADAU1701 through SigmaStudio

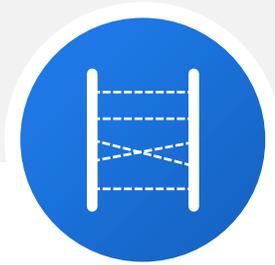
By Sure Electronics



# Overview

ADAU1701 is a high performance digital signal processor of Analog Devices Inc. We have developed many products integrated with ADAU1701, which support programming via SigmaStudio after connection with our programming boards.

This document is meant to show you how to program WONDOM ADAU1701 DSP products with SigmaStudio.



## Preparation



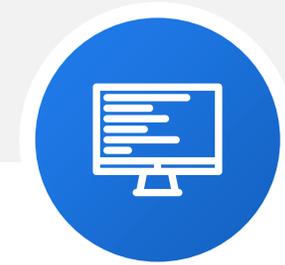
We need to know the products and software that we need before start. In this part, we will focus on introduction of WONDOM products integrated with DSP and ICP programmers.



## Configuration



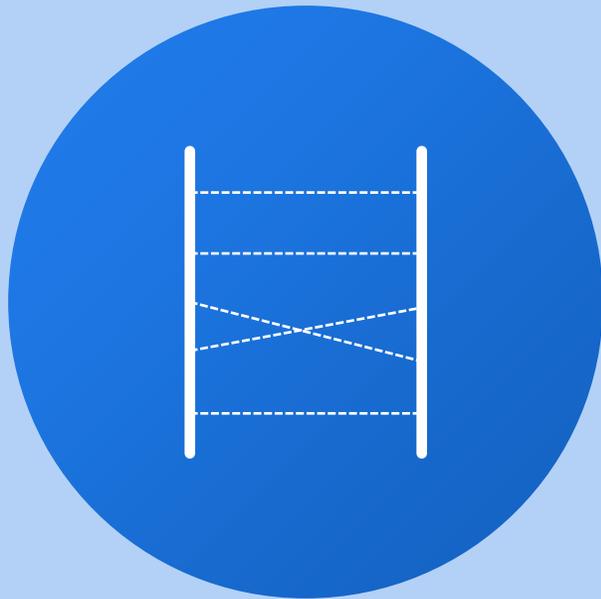
In this part, we will give you a brief introduction of SigmaStudio interface and some documents you may need.



## Examples



We will demonstrate to write basic programs based on WONDOM DSP products for your reference, so that you can get started quickly.



# Preparation

---

We need to know the products and software that we need before start.

In this part, we will focus on introduction of WONDOM products integrated with DSP and ICP programmers.

# Products List – with ADAU1701 DSP

The following are the products list of WONDOM products that are integrated with ADAU1701 DSP.

**The functions vary to different products. Please select to your requirements.**

Model	Description	SS Program	APP Control	PC UI Control
APM2	2-in, 4-out ADAU1701 DSP Kernel Board	Y	Y	Y
JAB3	Mono / Stereo Class D Audio Amplifier Board Integrated with ADAU1701 DSP	Y	Y	Y
JAB3+	Mono / Stereo Audio Amplifier Board Integrated with ADAU1701 DSP & BT V5.0	Y	N	Y
JAB4	4CH 30W Audio Amplifier Board Integrated with ADAU1701 DSP & BT V5.0, Supporting Configuration as 4.0 / 2.1 / 2.0 / 0.2 Output Mode	Y	N	Y
JAB5	4CH 100W Audio Amplifier Board Integrated with ADAU1701 DSP & BT V5.0, Supporting Configuration as 4.0 / 2.1 / 2.0 / 0.2 Output Mode	Y	N	Y

# Products List – ICP Series

Different ICP products support different functions. Please select to your requirements.

Model	Description	SS Program	APP Control	PC UI Control
ICP1	In-Circuit Programmer	Y	N	N
ICP3	In-Circuit Programmer Integrated with Bluetooth BLE	Y	Y	N
ICP5	In-Circuit Programmer Integrated with Bluetooth BLE & USB to UART	Y	Y*	Y

\* The function only works when both products with ADAU1701 DSP and the ICP board support it, i.e. JAB3+ doesn't support APP control, so even if you buy ICP3 or ICP5 that supports APP control, JAB3+ still cannot support APP control after connection.



# Software - SigmaStudio

If you never install SigmaStudio, you can click the following link for download and installation.

<https://www.analog.com/sigmastudioweb>

It's highly suggested that you remember the following link for official support. You can find algorithm information, tutorials, examples and other useful documentation as reference.

[SigmaStudio and SigmaDSP Documentation \[Analog Devices Wiki\]](https://www.analog.com/ADIWiki)

## SigmaStudio Toolbox

[Click here to return to the SigmaStudio and SigmaDSP Documentation top page.](#)

The Toolbox contains the building blocks for constructing a system design. The available blocks will depend on the DSP processor(s) used in the project.

Click a category to access detailed algorithm information. In most cases you can also select an algorithm in SigmaStudio, then press F1, to directly access its Wiki page.

- **System**
- **ADI Algorithms**
- **Advanced DSP**
  - Adaptive Mixer, Hilbert Transform
- **Basic DSP**
  - Delay, DSP Functions, Arithmetic Operations, Gain Cells, Logic, Index Lookup Tables
- **Counters**
- **Dynamics Processors**
  - Single-Band Compressors, Multi-Band Compressors, Limiters
- **Filters**
  - FIR Filters, Adaptive FIR Filters, Second Order Filters, Parametric Filters, Crossover, Miscellaneous Filters
- **Frequency Domain** (ADAU145x, ADAU146x, ADSP-2158x, and ADSP-SC58x only)
  - FFT, IFFT, Windows
- **GPIO Conditioning**
  - Pushbuttons, Volume Controls, Rotary Encoders, Debounce
- **Input / Output**
  - Input/Output from/to hardware: ASRC, SPDIF, GPIO, Serial Ports, Interface Write/Read
- **Installation Procedure**
- **Development Environment**
- **Using SigmaStudio**
- **SigmaDSP Architecture**
- **Supported ICs**
- **Release Information**

### Documentation

**Toolbox (Detailed Algorithm Information)**  
This section contains detailed

**Tutorials, Examples, and Documentation**  
This section contains tutorials

**Installation Procedure**  
This section describes how to install

**Development Environment**  
This section contains information

**Using SigmaStudio**  
This section describes the basic

**SigmaDSP Architecture**  
This section contains details about

**Supported ICs**  
This section describes the list of

**Release Information**  
This section contains features and

- **Frequency Domain** (ADAU145x, ADAU146x, ADSP-2158x)
  - FFT, IFFT, Windows
- **GPIO Conditioning**
  - Pushbuttons, Volume Controls, Rotary Encoder
- **Input / Output**
  - Input/Output from/to hardware: ASRC, SPDIF, GPIO
- **Level Detectors / Lookup Tables**
  - Lookup Tables, Signal Detect, RMS Level Meter
- **Licensed Algorithms**
- **Master Control Port** (ADAU145x and ADAU146x only)
  - I2C Writes, I2C Reads, SPI Writes, SPI Reads
- **Miscellaneous** (ADSP-2158x and ADSP-SC58x only)
  - PCM/PCMx Converters
- **Multi Rate Processing** (ADAU145x and ADAU146x only)
  - Upsample, Downsample, Synchronous SRC
- **Multiplexers / Demultiplexers**
  - State Machine, Demultiplexers, Multiplexers, Ctr
- **Mixers / Splitters**
  - Audio Signal Routers, Mixers, Splitters
- **Non-Linear Processors**
  - Assorted Clipping Algorithms (Soft, Hard, Asym)
- **Sources**
  - Chimes, DC Sources, Sweeps, Switches, Voltage
- **Volume Controls**
  - Gain Sliders, Mute, Envelopes

Table of Contents

- SigmaStudio and SigmaDSP Documentation
- Documentation Sections
- Getting Support
- Helpful Hints
- Feature Wishlist

Support was also added for SHARC to add quality digital signal processing while at the same time focusing on more be wired together, as in a schematic, and digital ones; yet it is powerful enough to costs without sacrificing quality or

you directly to its Wiki page.  
the left sidebar to navigate between topics.  
rZone support forum.



# Configuration

---

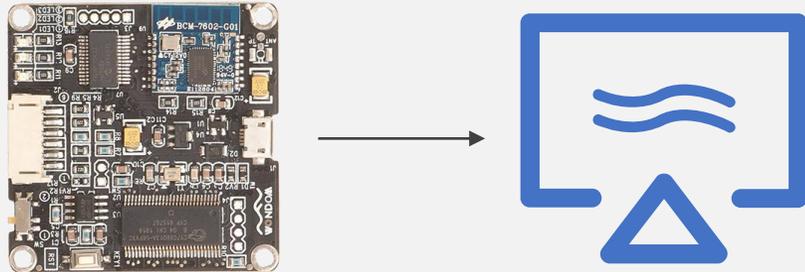
In this part, we will give you a brief introduction of SigmaStudio interface and some documents you may need.



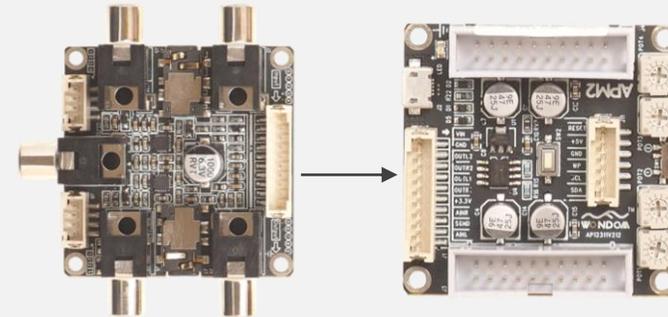
# Hardware Connection

We will take connection of APM2 and ICP3 as an example here.  
Connection based on other products are similar.

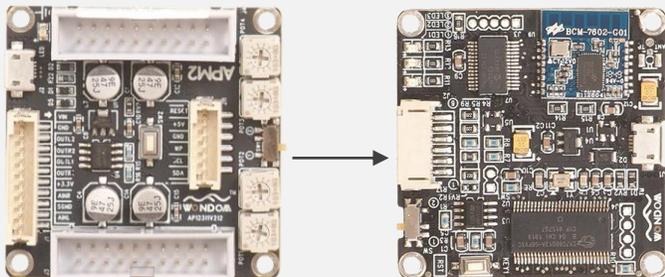
- ① Connect ICP3 to PC, run SigmaStudio, create a new project, drag “USBi”, “ADAU1701”, “E2Prom”, complete logic connection to see if ICP3 has been recognized successfully;



- ② Connect APM2 & APM3, connect audio source and speakers, power up the system



- ③ Connect APM2 with ICP3 through a 6-pin cable



- ④ Write program





# Brief Interface Introduction

The screenshot shows the SigmaStudio Hardware Configuration window. On the left is the TreeToolBox with a list of components under 'Processors (ICs / DSPs)'. The main workspace shows a schematic with a 'USB Interface' block connected to 'IC 1' (AD1701) and 'IC 2' (E2Prom). The 'Config' tab at the bottom shows parameters for 'IC 1 - 170x\140x Register Control' and 'IC 2 - WinE2PromLoader'. The 'Capture' window is empty, and the 'Output' window shows 'IC 1: Params' and 'IC 2: Params'. The status bar at the bottom indicates '100%' zoom and 'Design Mode'.

**Devices**  
ICs/DSPs & USB  
Communication  
module

**Hardware Configuration**  
Drag corresponding DSP & USB communication  
module and complete logic connection



# Brief Interface Introduction

The screenshot displays the SigmaStudio software interface. On the left, the **TreeToolBox** lists various components including Processors (ICs / DSPs) such as AD1940, ADAV4x, AD193x, AD195x, E2Pr om, ADAU1 701, ADAU1 702, ADAU1 401, ADAU1 44x, ADAV4x, ADAV46xx, ADAU1 76x, ADAU1 461, ADAU1 361, ADAU1 961, ADAU1 701, ADAU1 702, ADAU1 772, ADAU1 966, ADAU1 451, ADAU1 450, SSM3625, ADAU1 462, ADAU1 463, ADAU1 467, ADAU1 777, ADAU1 777, and SSM2529. The main workspace shows the **Hardware Configuration** section for the **IC 1 - 170x\140x Register Control**. It includes a **DSP Core** block with settings for **Program Length** (1x (1024 Instructi...)) and **RAM Modulo** (8). There are also **Serial Input** and **Serial Output 1 (channels 0-7)** blocks with various configuration options like **LRCLK polarity**, **Frame Sync Type**, **MSB Position**, and **Word length**. A **GPIO** table is visible, showing pins MP0 through MP11, all configured as **Input** with **GPIO Debounce** and **Inv** checkboxes. The **Output** section on the right shows **Action Output** and **Action**. The bottom status bar indicates **100%** and **Design Mode**.

**Devices**  
ICs/DSPs & USB  
Communication  
module

**Hardware Register**  
This page appears when ADAU1701 is  
dragged into hardware configuration section.  
GPIOs are set here.



# Brief Interface Introduction

The screenshot displays the SigmaStudio software interface. On the left is the **TreeToolbox** containing various components like **Simulation Probe**, **Simulation Stimuli**, **ADI Algorithms**, **Advanced DSP**, **Basic DSP**, **Counters**, **Custom Algorithms**, **Dynamics Processors**, **GPIO**, **Level Detectors/Lookup Tables**, **Mixers/Splitters**, **Muxes/Demuxes**, **Volume Controls**, and **Single/Multiple Controls**. The main workspace shows a **Schematic** design with an **Input1** block connected to a **Single 1** block, which is then connected to two **DAC0** and **DAC1** blocks. The **Output** panel on the right shows **Action Output** and **Action**. The **Capture** panel at the bottom has a table with columns: **Mode**, **Time**, **Cell Name**, **Parameter Name**, **Address**, **Value**, **Data**, and **Bytes**. The **Output** section shows **IC 1: Params** and **IC 2: Params**. The status bar at the bottom right indicates **100%** and **Design Mode**.

**Schematic  
Toolbox  
Algorithms  
& Modules**

**Schematic Design**

Drag desired algorithm here and complete the logic connection

Mode	Time	Cell Name	Parameter Name	Address	Value	Data	Bytes



# DSP Products Pin Definition

WONDOM products integrated with ADAU1701 DSP are equipped with terminal interfaces for various functions. You can get the detailed introduction in datasheet. Please find the documentation in the detailed product page on our website.

Model	Document
APM2	<a href="#">AA-AP23122 ADAU1701 Kernel Board</a>
JAB3	<a href="#">Audio Amplifier Boards Integrated with ADAU1701 DSP</a>
JAB3+	<a href="#">Audio Amplifier Boards Integrated with ADAU1701 DSP &amp; Bluetooth V5.0</a>
JAB4	<a href="#">4CH 30W Audio Amplifier Boards Integrated with ADAU1701 DSP &amp; Bluetooth V5.0</a>
JAB5	<a href="#">4CH 100W Audio Amplifier Boards Integrated with ADAU1701 DSP &amp; Bluetooth V5.0</a>



# ICP Programmer Pin Definition

ICP series is self-developed programming board, which can be used with WONDOM DSP products. After connection with ICP programmer, DSP products can support programming and remote control (It's up to specific product). Please find the user guide in the detailed product page on our website.

Model	Document	Video
ICP1	<a href="#">WODNOM ICP1 User Guide - Programming</a>	-
ICP3	<a href="#">WODNOM ICP3 User Guide – Programming &amp; APP Control</a>	<a href="#">Watch</a>
ICP5	<a href="#">WODNOM ICP5 User Guide – Programming, APP &amp; PC UI Control</a>	-



# Demo Programs for Programming

Here are the demo programs for the products. You can download them for reference. The demo program is only for demonstration of signal flow chart.

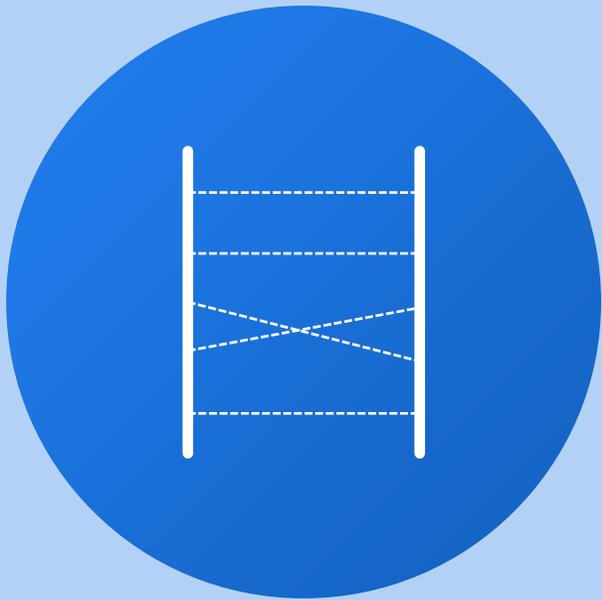
Products	Demo Program	Document you may need
APM2	<a href="#">APM2_SigmaStudio.dspproj</a>	<a href="#">Download</a>
JAB3 - Mono	<a href="#">JAB3_SigmaStudio_MONO.dspproj</a>	<a href="#">Download</a>
JAB3 - Stereo	<a href="#">JAB3_SigmaStudio_STEREO.dspproj</a>	
JAB3+	<a href="#">JAB3+_Stereo_ADAU1701_DEMOProgram.dspproj</a>	<a href="#">Download</a>
JAB4	<a href="#">JAB4_ADAU1701_DEMOProgram.dspproj</a>	-
JAB5	<a href="#">JAB5_ADAU1701_DEMOProgram.dspproj</a>	<a href="#">Download</a>



# Resources Correspondence

WONDOM products are developed based on ADAU1701 DSP. We have made use of ADAU1701 resources to provide basic functions like audio input & output, control. Therefore, it's necessary for us to understand the correspondence relationship between the hardware and program for further development.

Model	Document	Video
APM2	<a href="#">The correspondence between APM2 hardware and program</a>	-
JAB3	<a href="#">How to develop JAB3 &amp; Integrated ADAU1701 DSP</a>	<a href="#">Watch</a>
JAB3+	<a href="#">How to program JAB3+ through SigmaStudio</a>	<a href="#">Watch</a>
JAB5	<a href="#">How to program JAB5 through SigmaStudio</a>	<a href="#">Watch</a>



# Examples

---

We will demonstrate to write basic programs based on WONDOM DSP products for your reference, so that you can get started quickly.

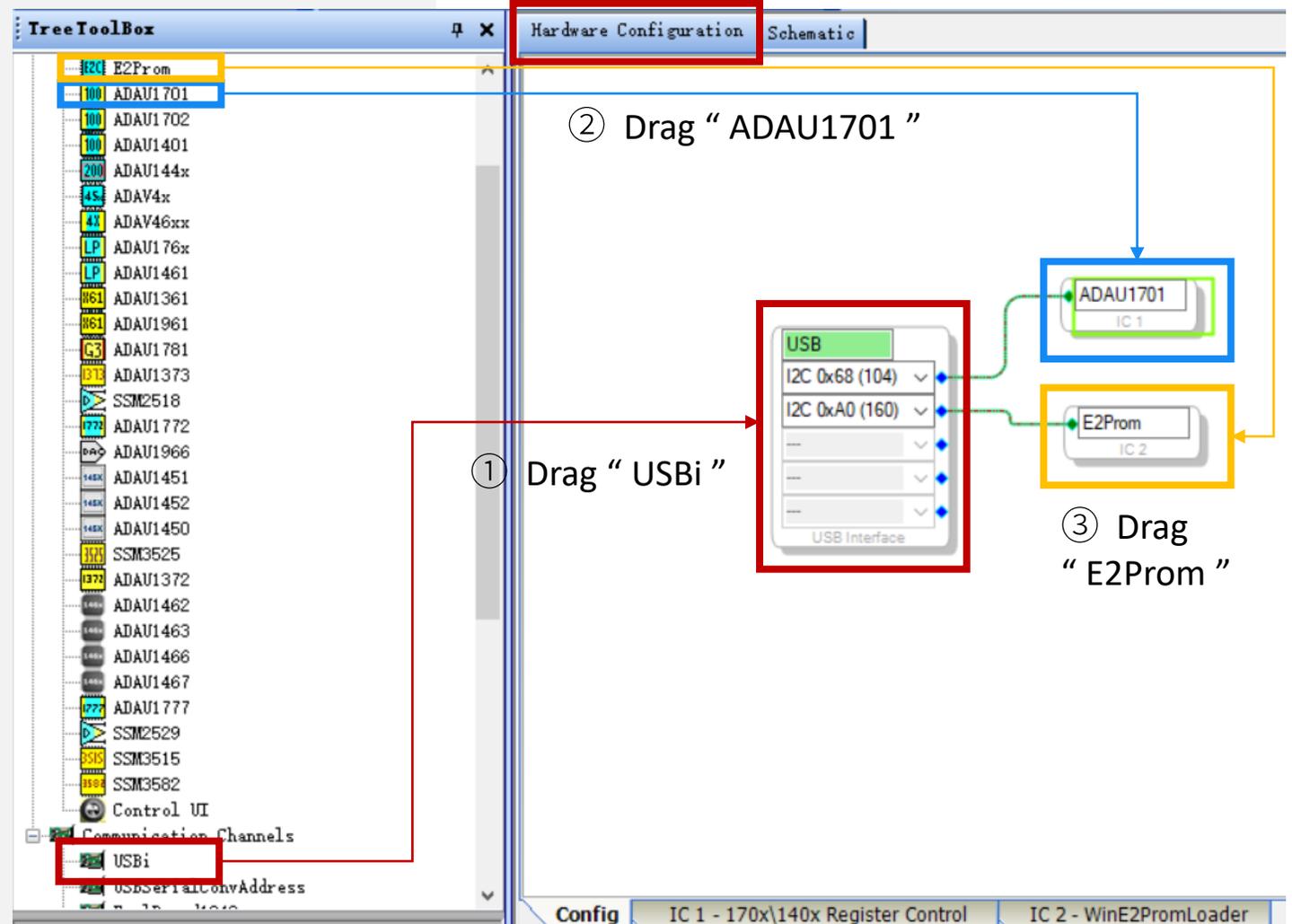


# Basic Program

First step, double click to run SigmaStudio. Then click “File”-->“New Project” to create a new project.

Second step, after connecting ICP programmer to PC, find “USBi”、 “ADAU1701”、 “E2Prom” in “TreeToolBox” and then drag them into “config” section. Please pay attention to order to ensure that ADAU1701 shows IC1 and E2Prom shows IC2.

If USBi displays in green, it means ICP has been recognized successfully; If it’s orange, please try to reconnect ICP.

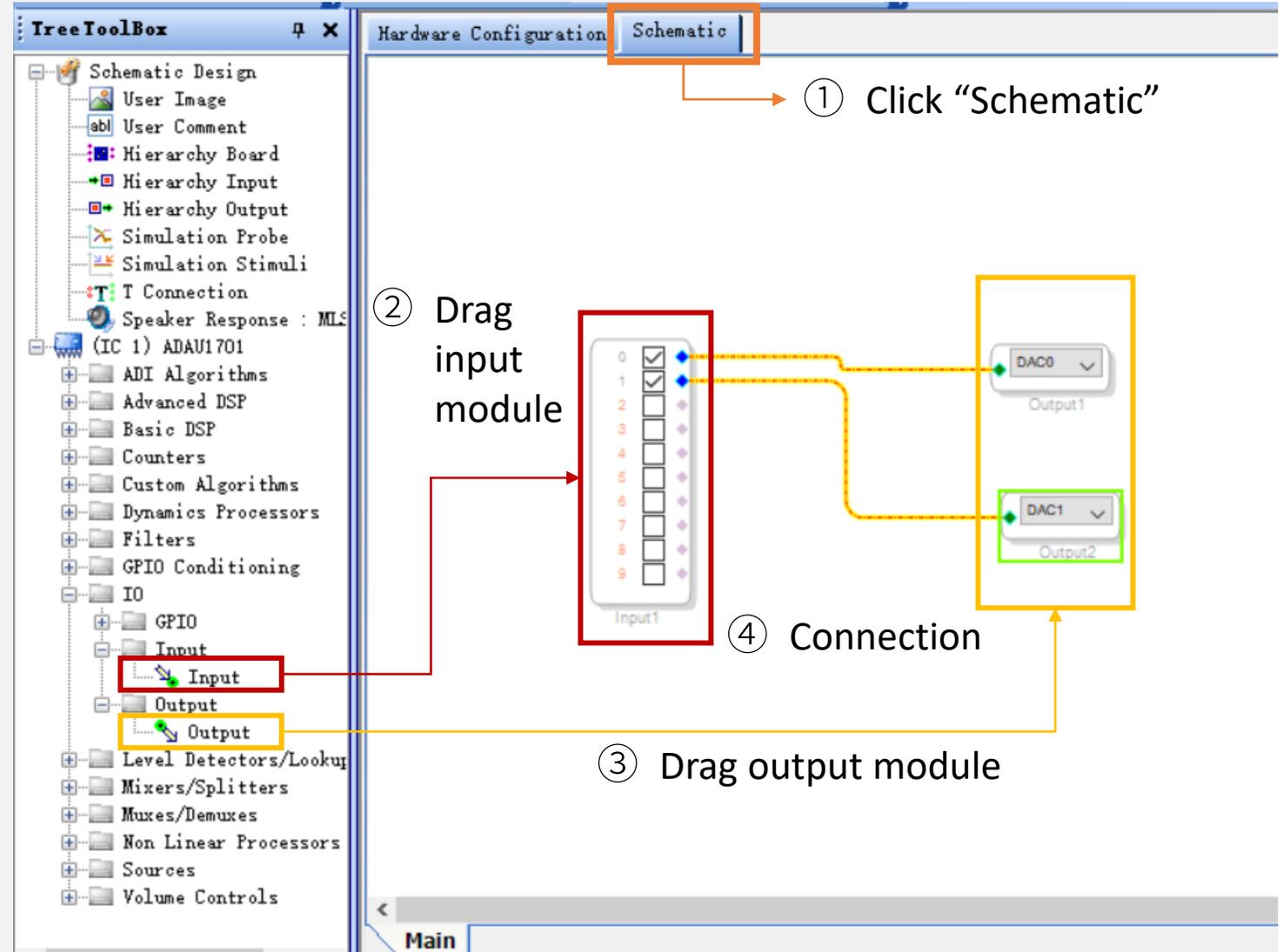




# Basic Program – Bypass

Let's write a bypass program at first.

- ① Click "Schematic" to enter schematic design section
- ② Drag "IO"->"Input"->"Input" to design section from "TreeToolBox". We can see 0 & 1 of input module are ticked. They are for analog input.
- ③ Drag "IO"->"Output"->"Output" . We want stereo output, so we need to drag two output modules. Stereo output on APM3 are corresponded with DAC0 & DAC1.
- ④ Complete logic connection between input and output modules.



(编写程序)



# Basic Program – Bypass

Download finished program into DSP. Please note, here, download refers to online simulation under debug mode. The program will be lost once the power is off. If you need to operate offline, please refer to “Program Writing” chapter.

- ⑤ Click “Link Compile Download” on the menu to download program into ADAU1701, as shown below
- ⑥ If it shows “Active: Downloaded” at the bottom right corner, program is downloaded successfully
- ⑦ Play music to see if there is music playing from speakers



(Program Download)

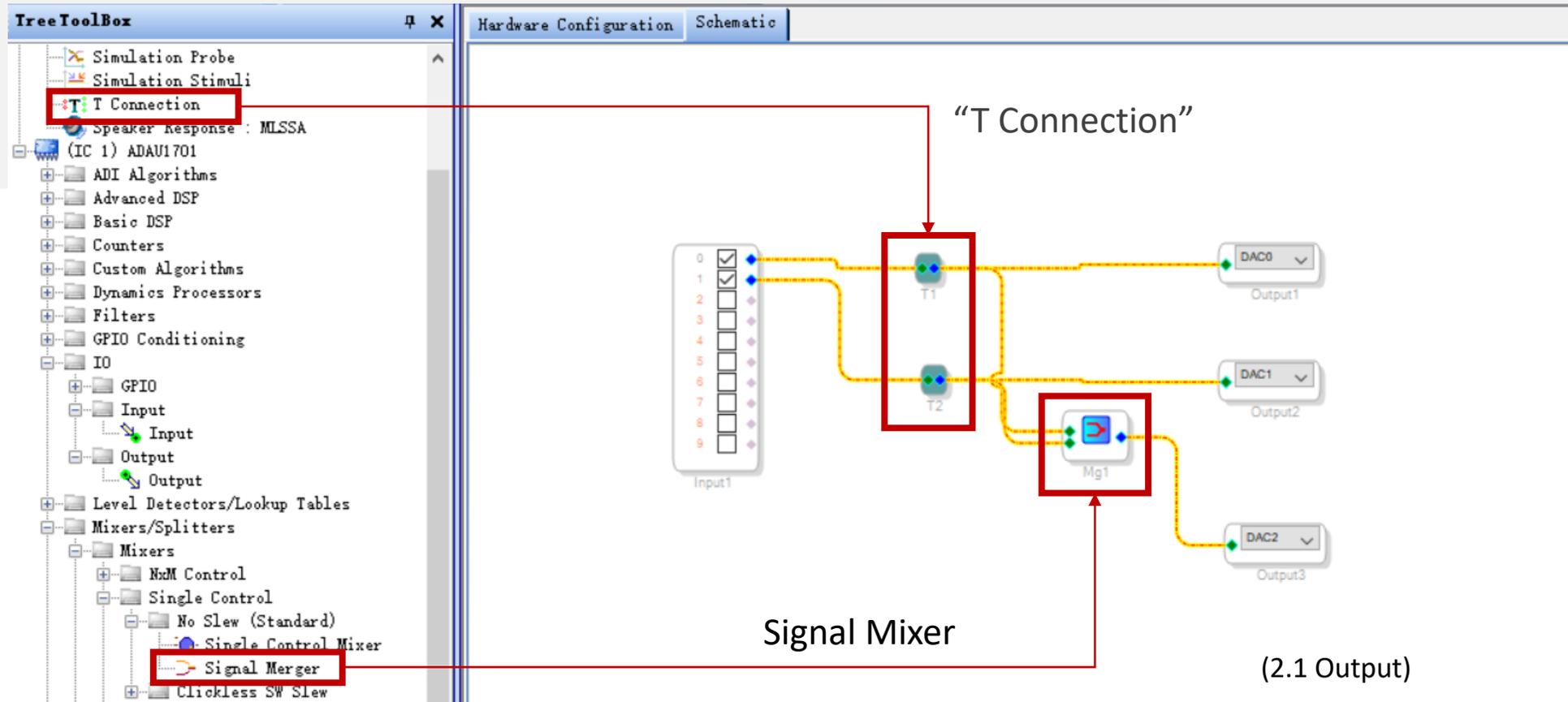


(Download Success)



# Basic Program – Bypass

APM2 is the kernel board and APM3 is the matched 2-in, 3-out interface board. In last example, we write stereo output. What if we want subwoofer output for 2.1 system? We can drag another output module, then mix left and right channel.

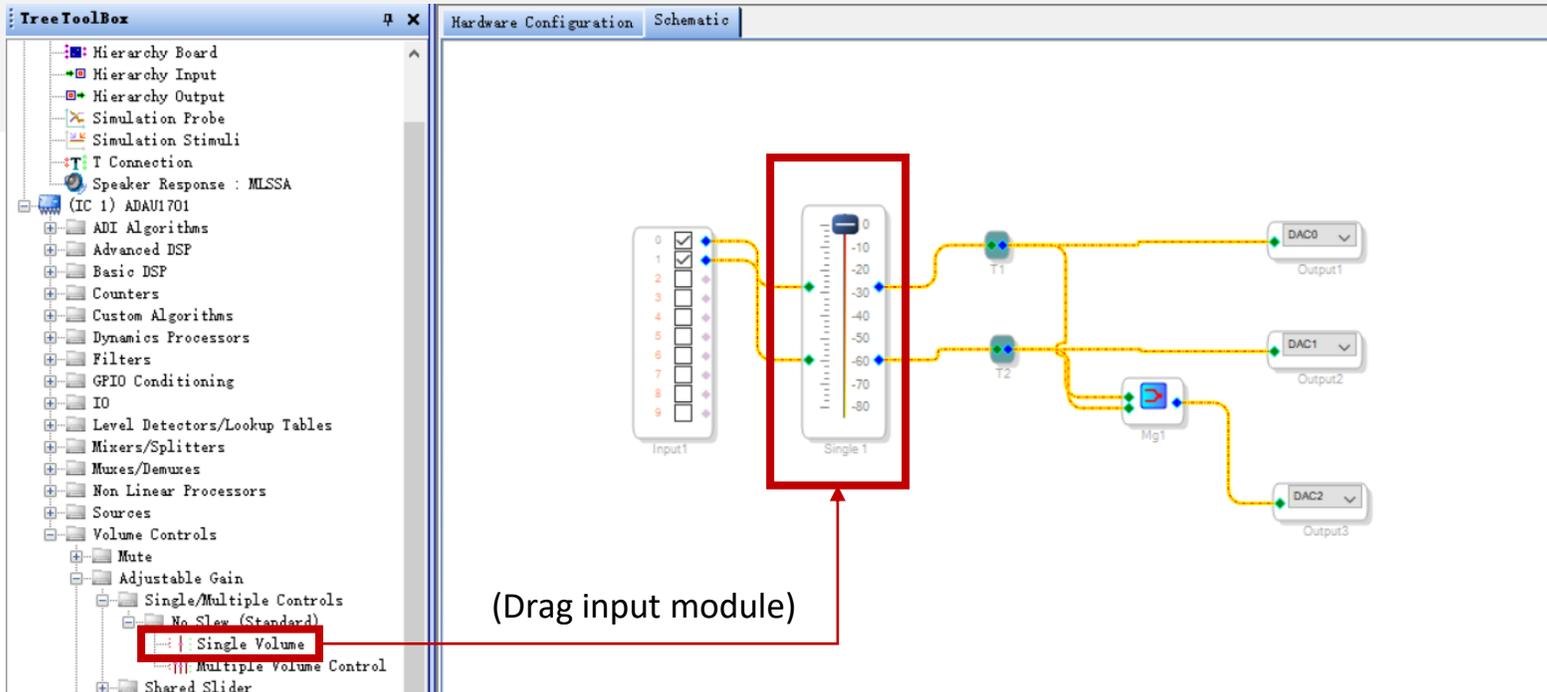




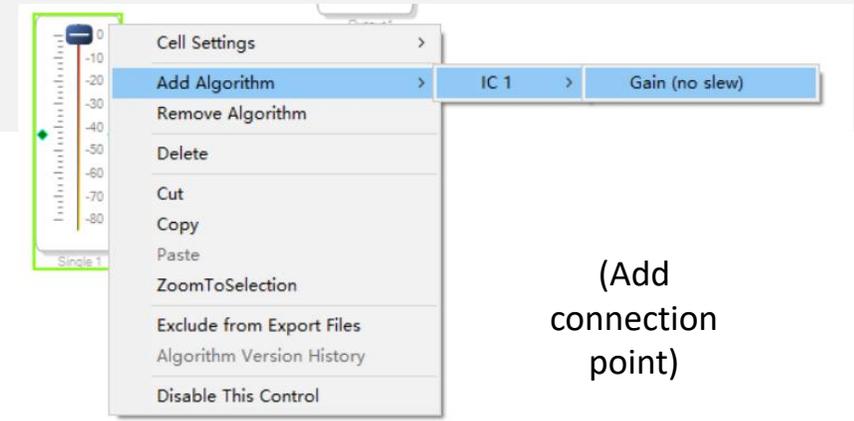
# Basic Program – Volume Control

How to adjust overall volume? We need to make use of “Volume Controls” modules. There are many functions in this module group. We will use one for demonstration only. You can explore other modules on your own.

Put volume control module into design section as below. You can adjust overall volume by dragging the slider. Put cursor on this module, then right click without selecting it, you will find detailed settings of volume control, such as max value, steps, etc.



(Drag input module)



(Add connection point)

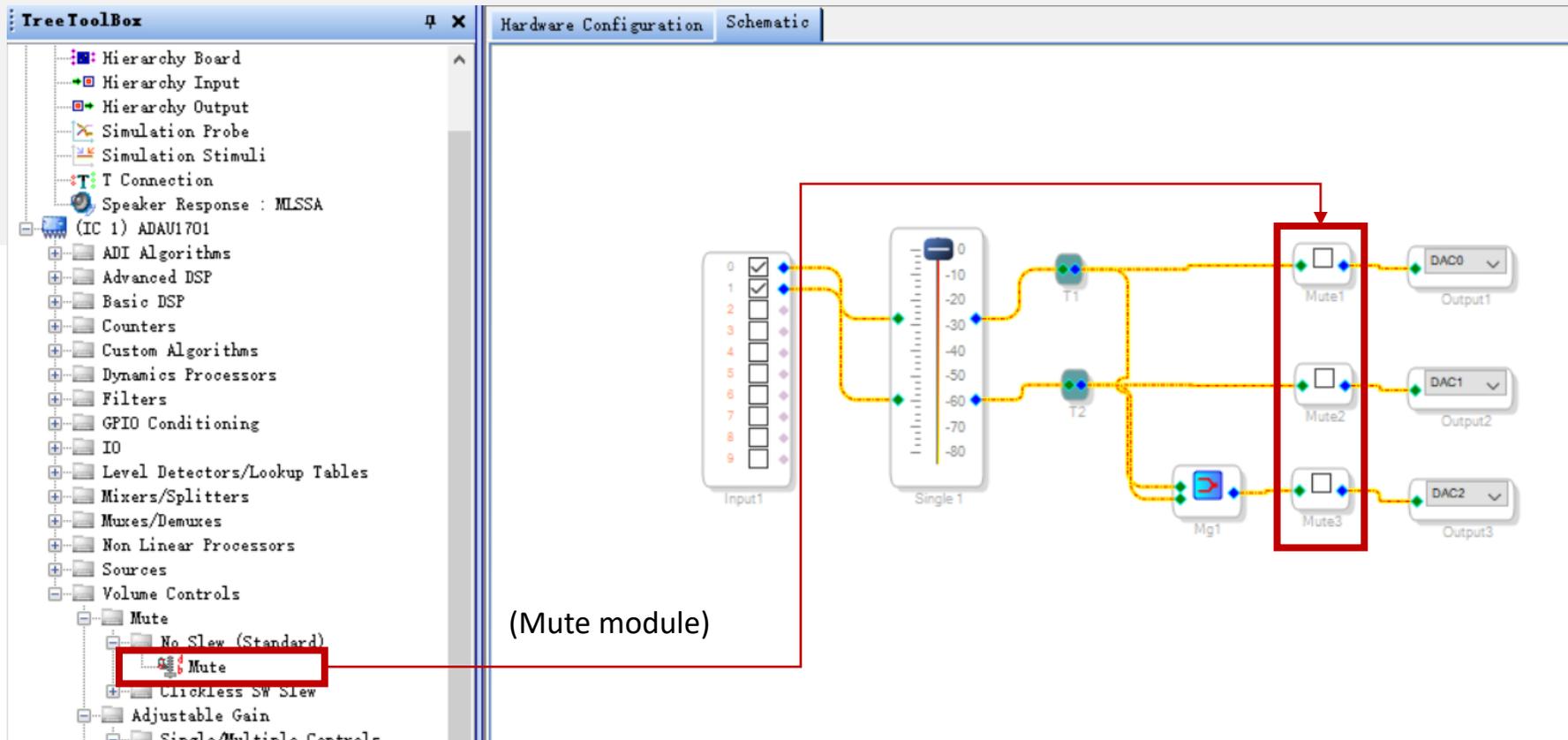
There is only one connect point in input module. Select the module, right click and add points.



# Basic Program – Volume Control

I want to control whether a channel is mute or unmute, what should I do?

This function can be realized by “Mute” module in the “Volume Controls” group. The connection is as follows.



If the mute module is unticked, this channel plays music normally.

If it's ticked, this channel will be mute.

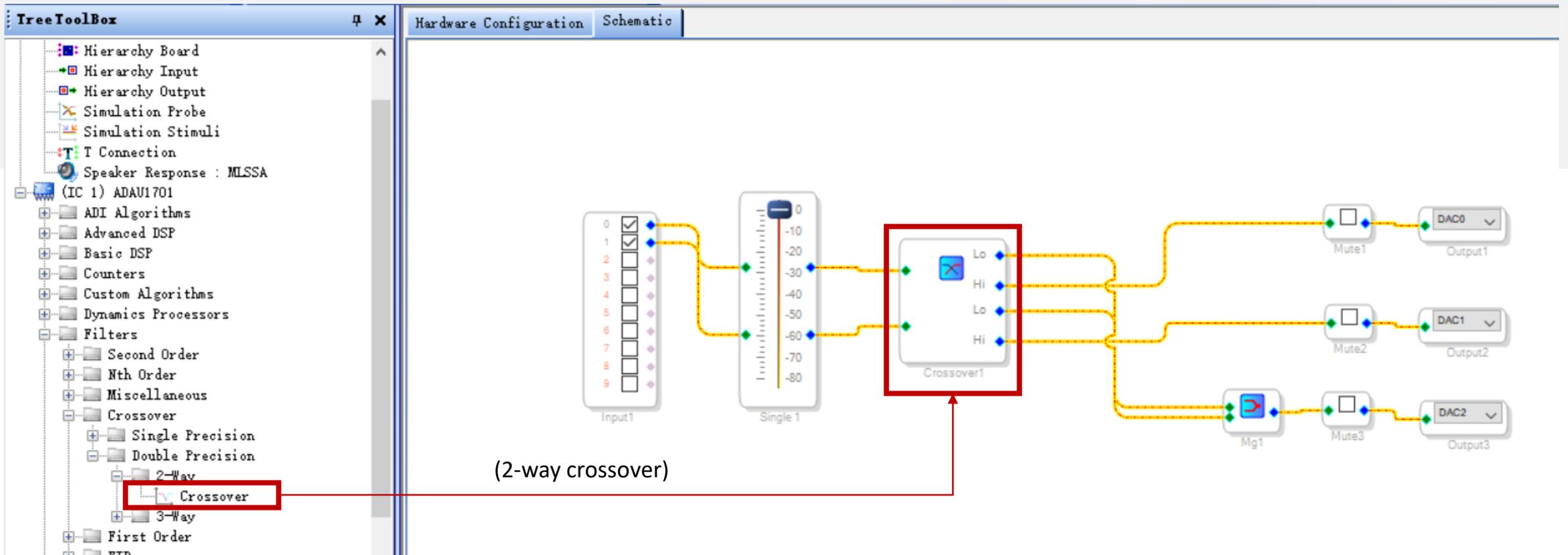
It's usually used to check if each channel works.



# Basic Program – Crossover

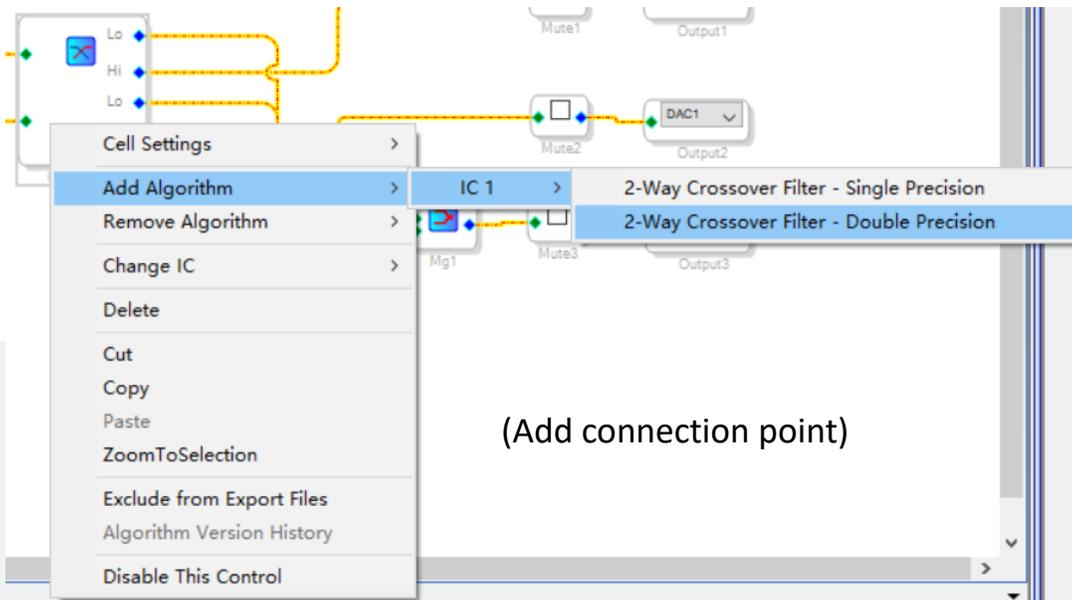
This time, let's add crossover function. We employ “Filters”-“Crossover”-“Double Precision”-“2 -Way”-“Crossover” module here. You can try other modules by yourself.

Signal Flow: Input left and right channel signal, go through volume control, each channel is split into high frequency and low frequency. The high is transmitted to stereo output while the low is mixed and transferred to subwoofer output.

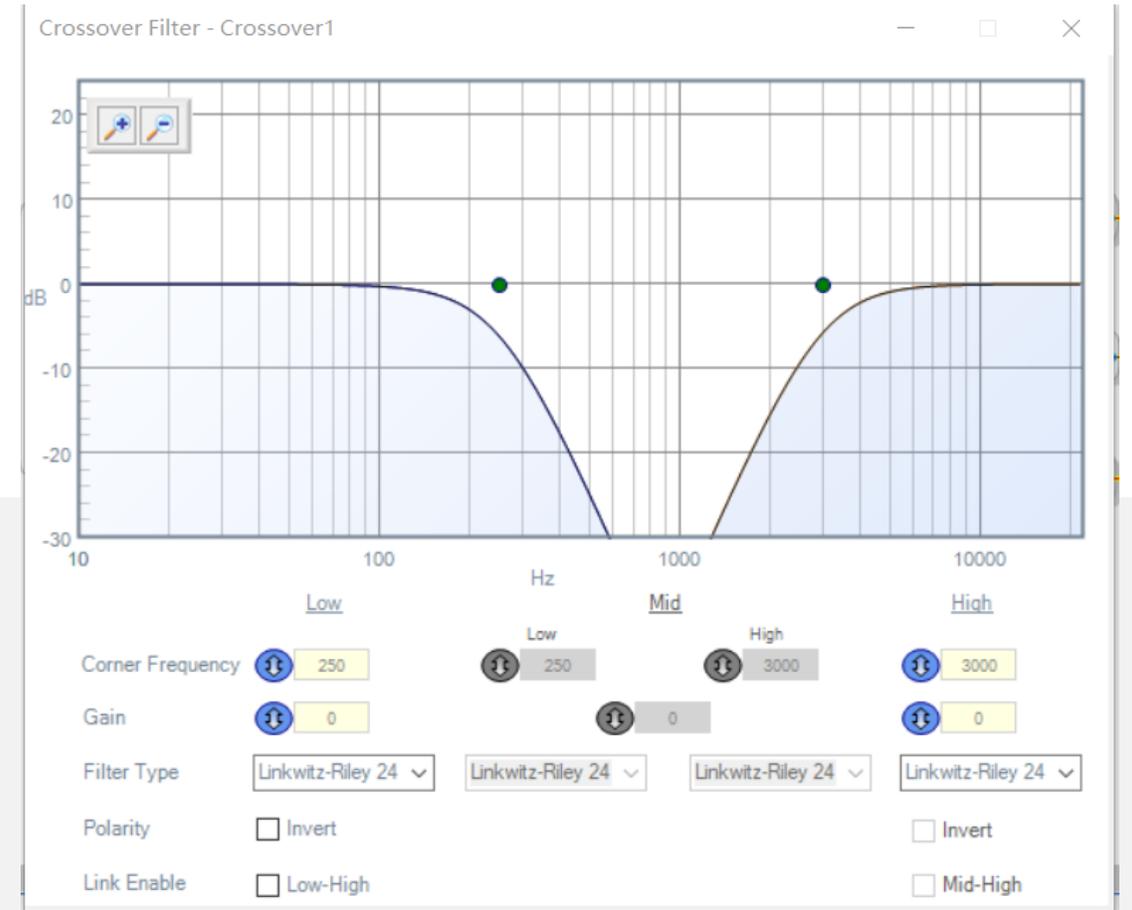




# Basic Program – Crossover



If crossover module doesn't have enough the connection points, you can select the module and right click, then choose "Add Algorithm".

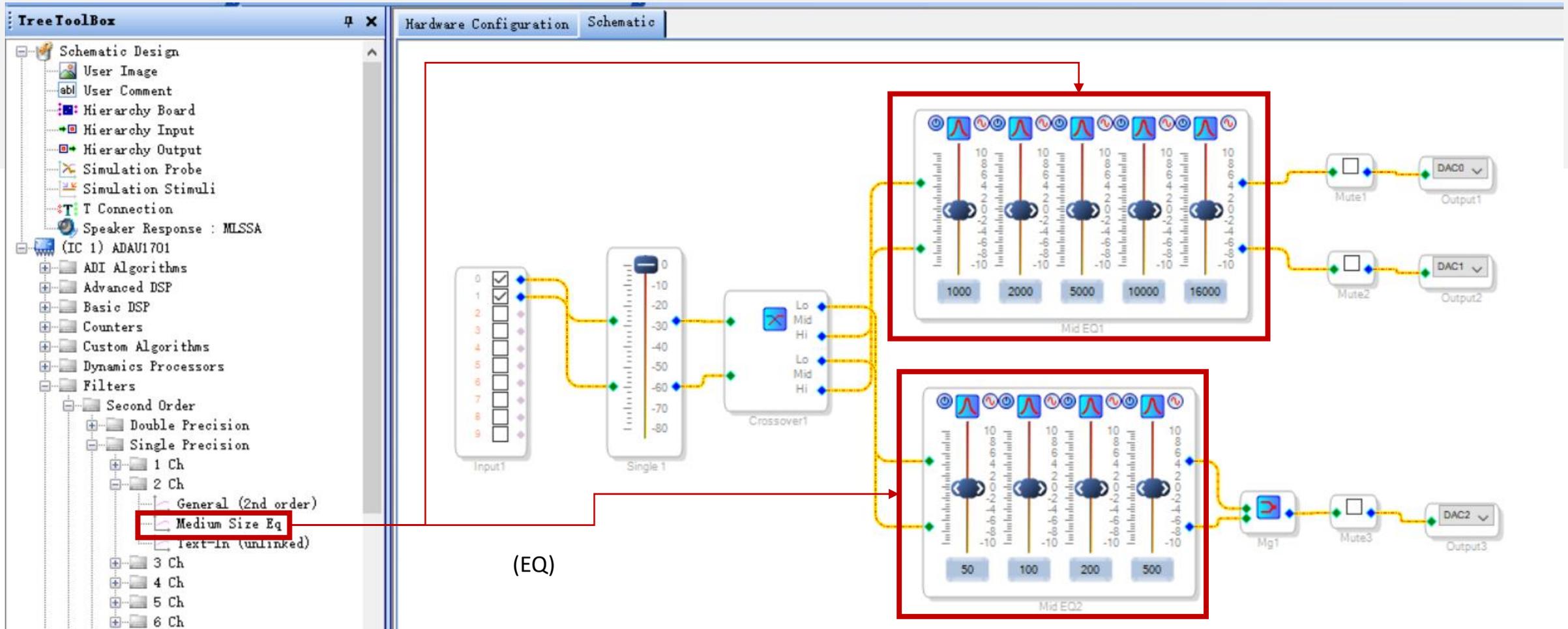


Click the curve icon of the crossover module, you can see detailed settings. You can adjust cut-off frequency, gain, filter type and polarity.



# Basic Program – EQ

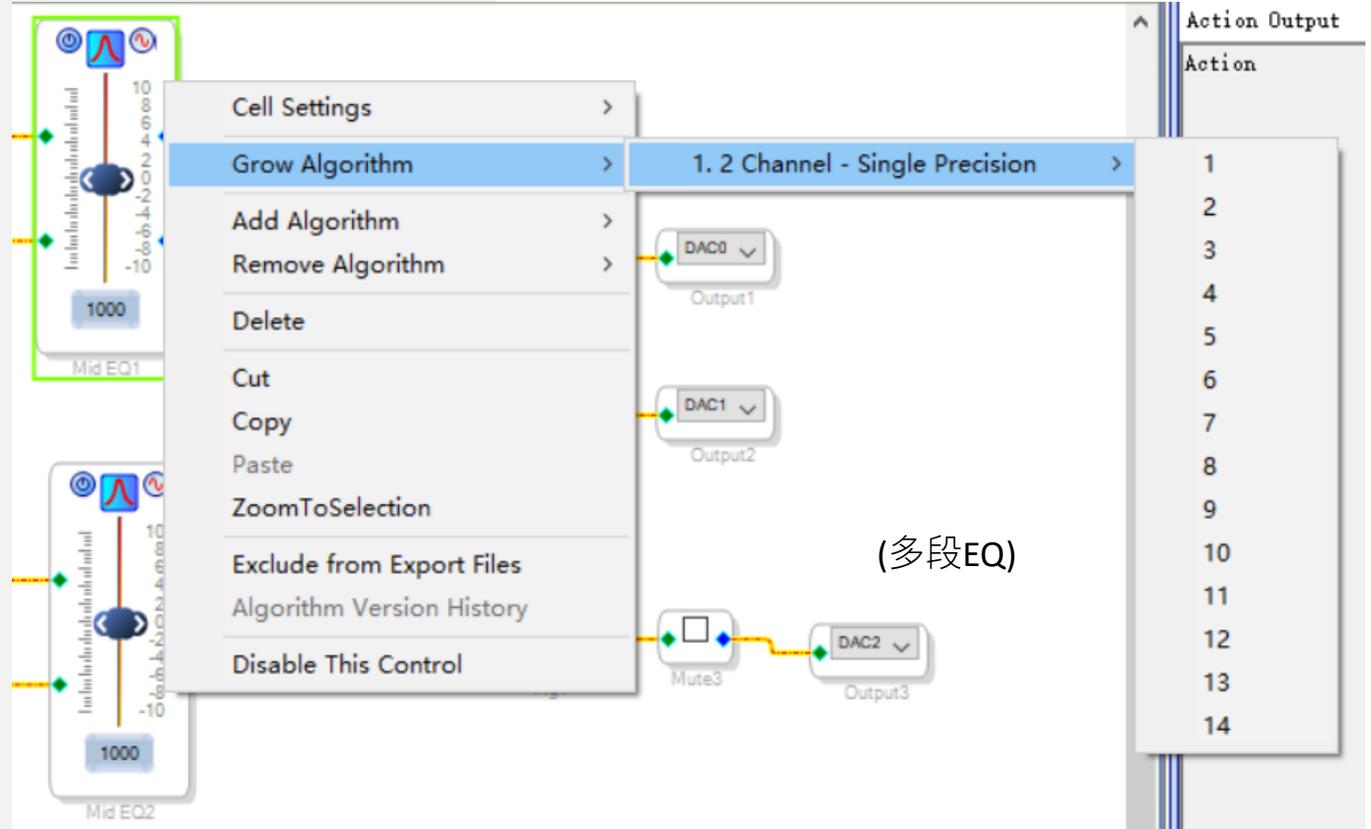
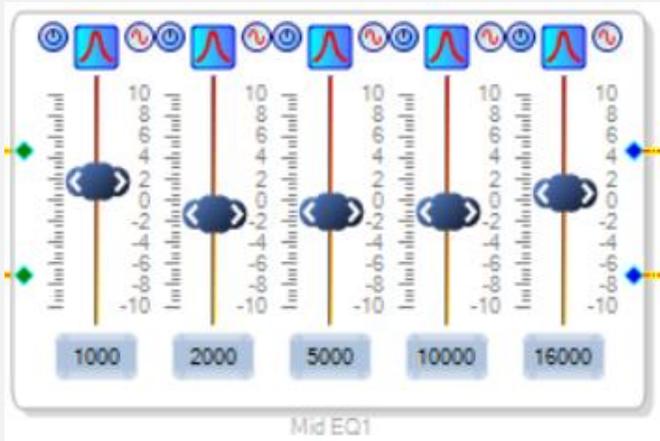
Now, we want to add EQ based on finished program, through which we can adjust gain of each frequency to achieve desired effects. Here we use “Filters”-“Second Order”-“Single Precision”-“2ch”-“Medium Size Eq” module.





# Basic Program – EQ

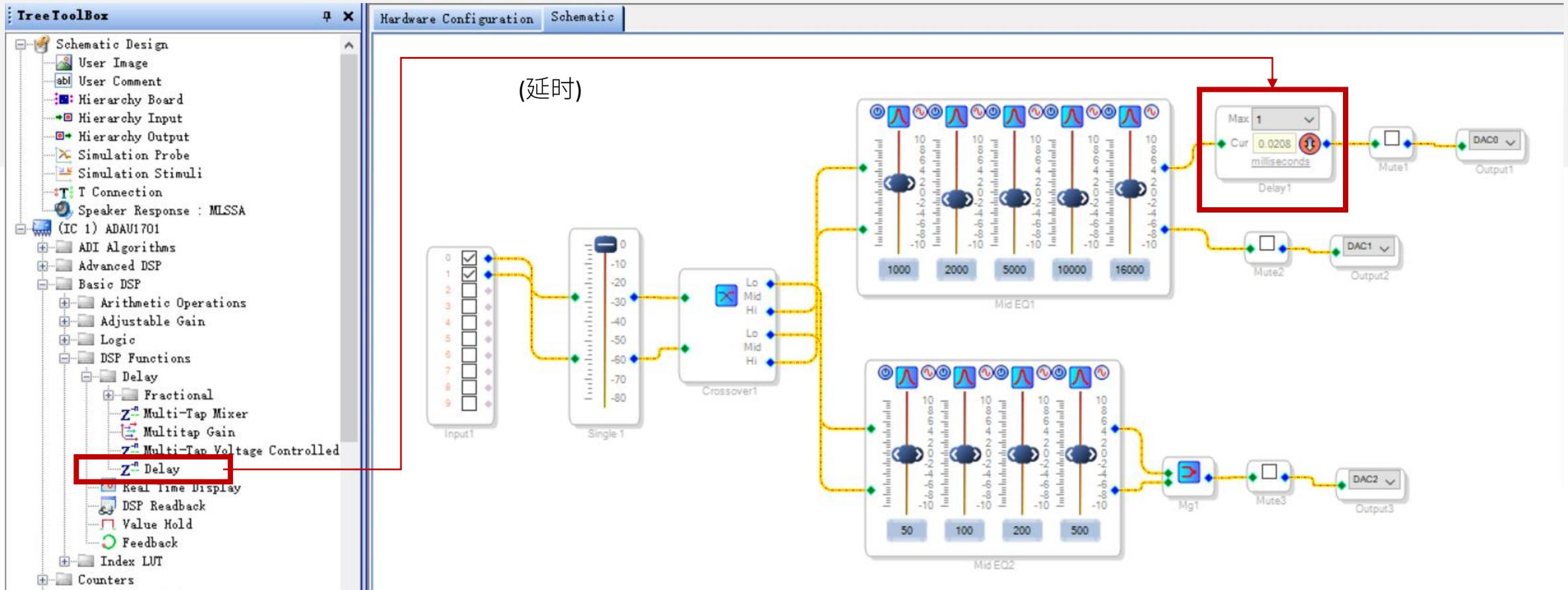
There is only one frequency band of the module. We can right click with it selected and choose how many bands you want. Each center frequency and gain are adjustable. Here is just for example.





# Basic Program – Delay

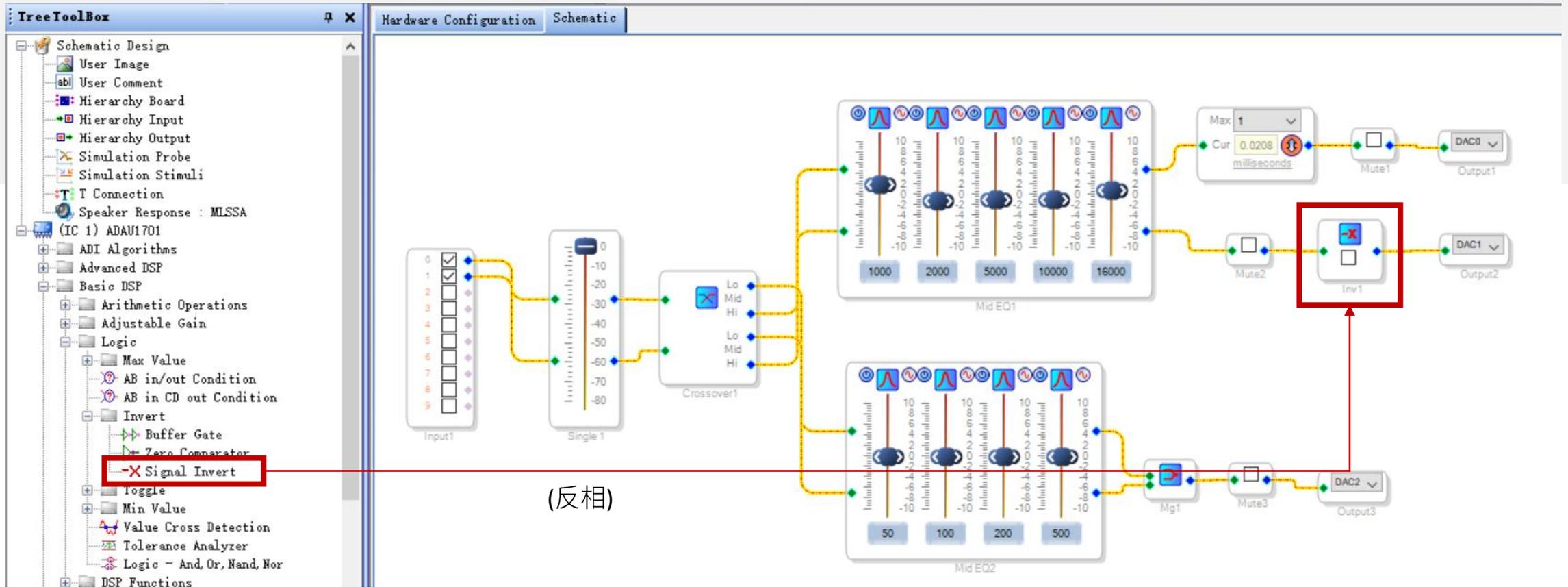
If you can hear apparent time difference when listening to music, you can use delay module to improve listening experience. We employ “Basic DSP”-“DSP Functions”-“Delay”-“Delay” module here。 Click “Samples” of “Delay” module to switch delay unit to “milliseconds”, which will be more convenient.





# Basic Program – Signal Invert

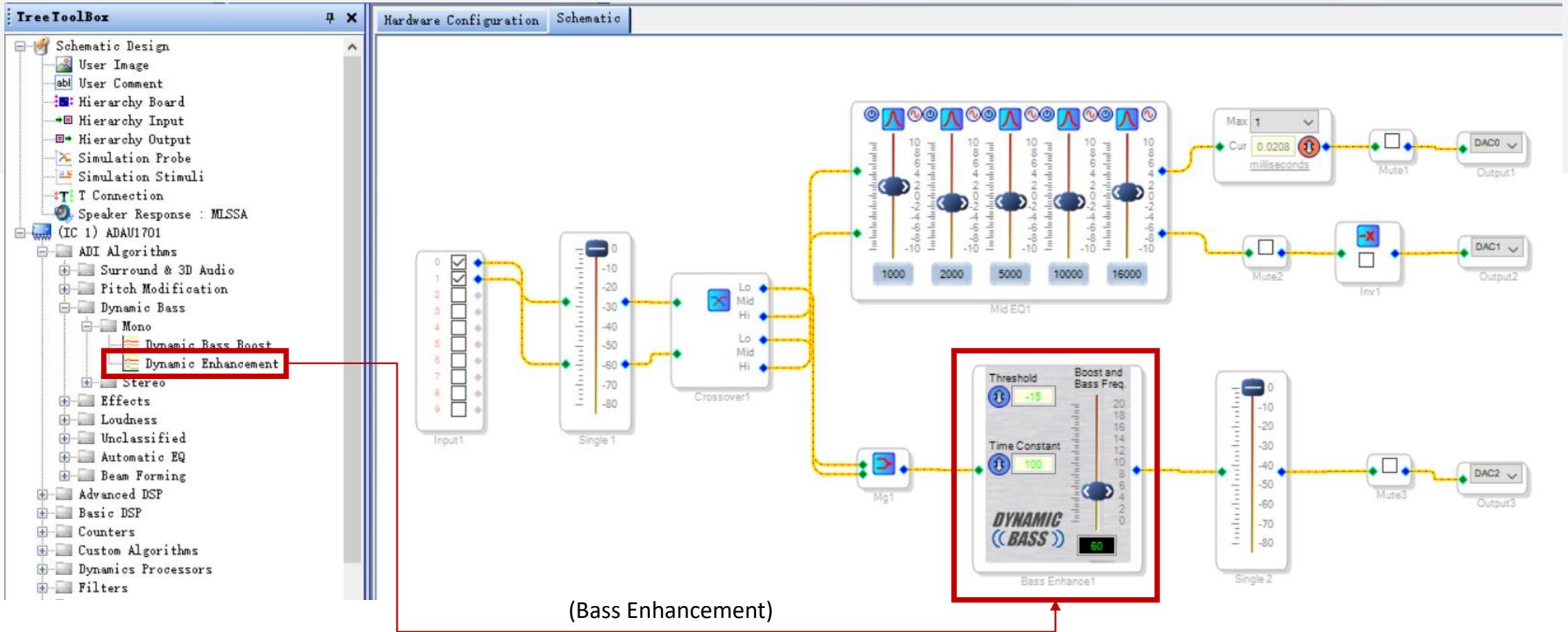
We may connect speakers reversely by accident and sometimes we don't want to reconnect. Then, we can use signal invert module. We use "Basic DSP"- "Logic"- "Invert"- "Signal Invert" module. Signal will be inverted after ticked.





# Basic Program – Bass Enhancement

Many customers want bass enhancement. We can get this function through “ADI Algorithms”-“Dynamic Bass” module. You can set the parameters according to the requirements of your audio project.

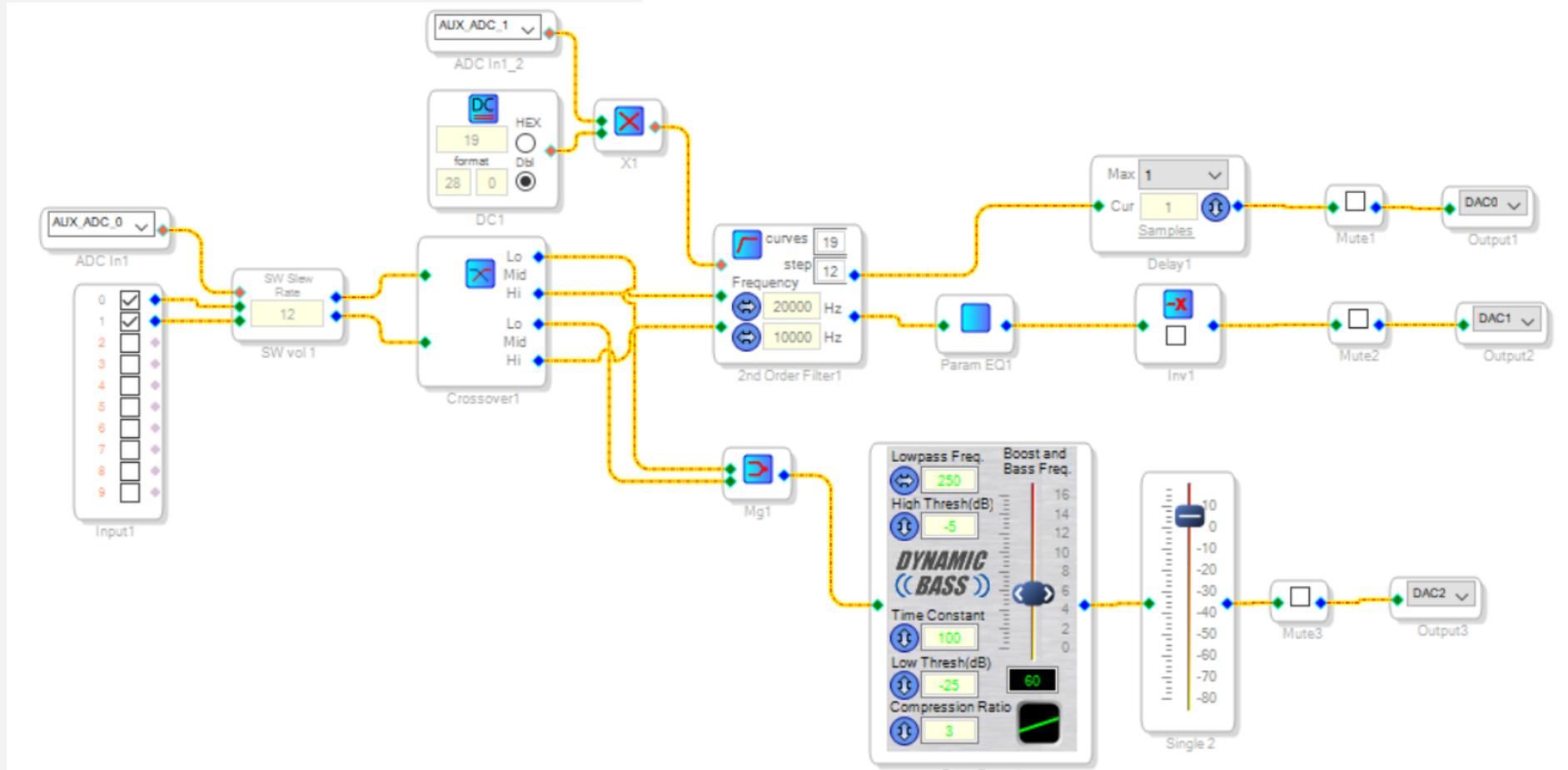




# Basic Program – Potentiometers

APM2 provides four on-board potentiometers by extending resources of ADAU1701 DSP. You can use them for desired functions. Here is just for example.

Please note, if you add potentiometer modules, beside logic connection in schematic design section, you need to configure GPIO in hardware register section.

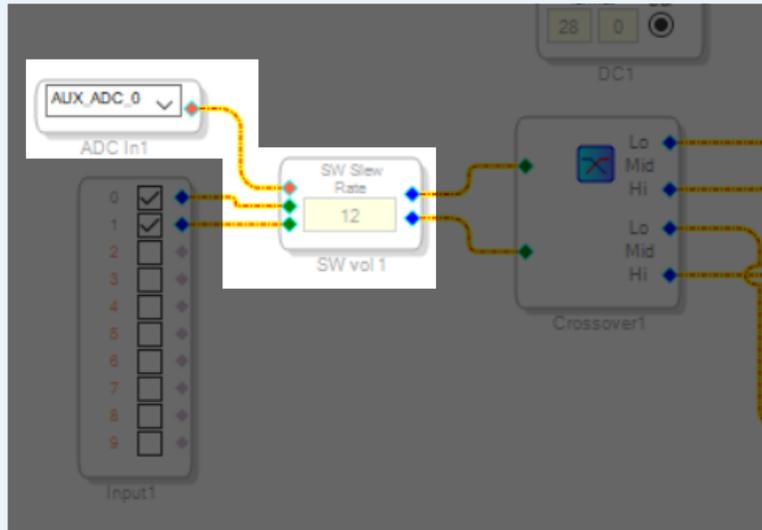




# Basic Program – Potentiometers

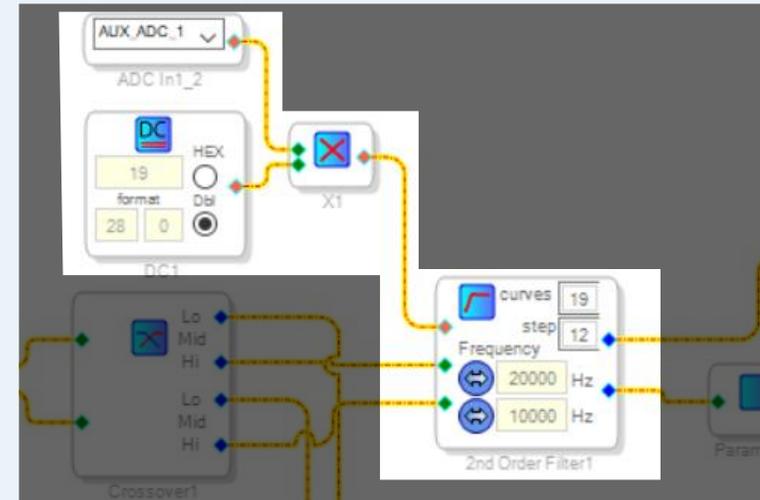
The potentiometers need to read hardware parameters then transmit to related modules for signal processing. Therefore, we need to adopt modules with control parameters.

## ① Overall Gain Adjustment



Employ gain control module with control parameters, drag “Auxiliary ADC Input” module, and complete logic connection

## ② High-pass Filter



This potentiometer works as high-pass filter of stereo. The filter module divides 10k-20kHz frequency into 19 curves. The potentiometer reads hardware voltage and multiply with the value of DC input Block to convert into frequency signal, which is transferred to filter to choose corresponding curve.



# Basic Program – Potentiometers

After schematic is done, we need to set GPIO in hardware register section. Otherwise, the potentiometers won't work normally.

Find the GPIO and choose the corresponding ADC in the drop-down menu.

Pin	Value	Direction	Inv
MP0	Low	Input GPIO Debounce	<input type="checkbox"/>
MP1	Low	Input GPIO Debounce	<input type="checkbox"/>
MP2	Low	Input GPIO Debounce	<input type="checkbox"/>
MP3	Low	Input GPIO Debounce	<input type="checkbox"/>
MP4	Low	Input GPIO No Debounce	<input type="checkbox"/>
MP5	Low	Output GPIO	<input type="checkbox"/>
MP6	Low	Output GPIO Open Collector	<input type="checkbox"/>
MP7	Low	Input Sdata_in2	<input type="checkbox"/>
MP8	Low	NA	<input type="checkbox"/>
MP9	Low	NA	<input type="checkbox"/>
MP10	Low	NA	<input type="checkbox"/>
MP11	Low	NA	<input type="checkbox"/>

Hardware Configuration Schematic

GPIO

Pin	Value	Direction	Inv
MP0	Low	Input GPIO Debounce	<input type="checkbox"/>
MP1	Low	Input GPIO Debounce	<input type="checkbox"/>
MP2	Low	ADC1	<input checked="" type="checkbox"/>
MP3	Low	Input GPIO Debounce	<input type="checkbox"/>
MP4	Low	Input GPIO Debounce	<input type="checkbox"/>
MP5	Low	Input GPIO Debounce	<input type="checkbox"/>
MP6	Low	Input GPIO Debounce	<input type="checkbox"/>
MP7	Low	Input GPIO Debounce	<input type="checkbox"/>
MP8	Low	Input GPIO Debounce	<input checked="" type="checkbox"/>
MP9	Low	ADC0	<input checked="" type="checkbox"/>
MP10	Low	Input GPIO Debounce	<input type="checkbox"/>
MP11	Low	Input GPIO Debounce	<input type="checkbox"/>

Control ADC

Enable

Input Filter: 4-bit Hyste

Adc	Value
Adc0	b 00000000
Adc1	b 00000000
Adc2	b 00000000
Adc3	b 00000000

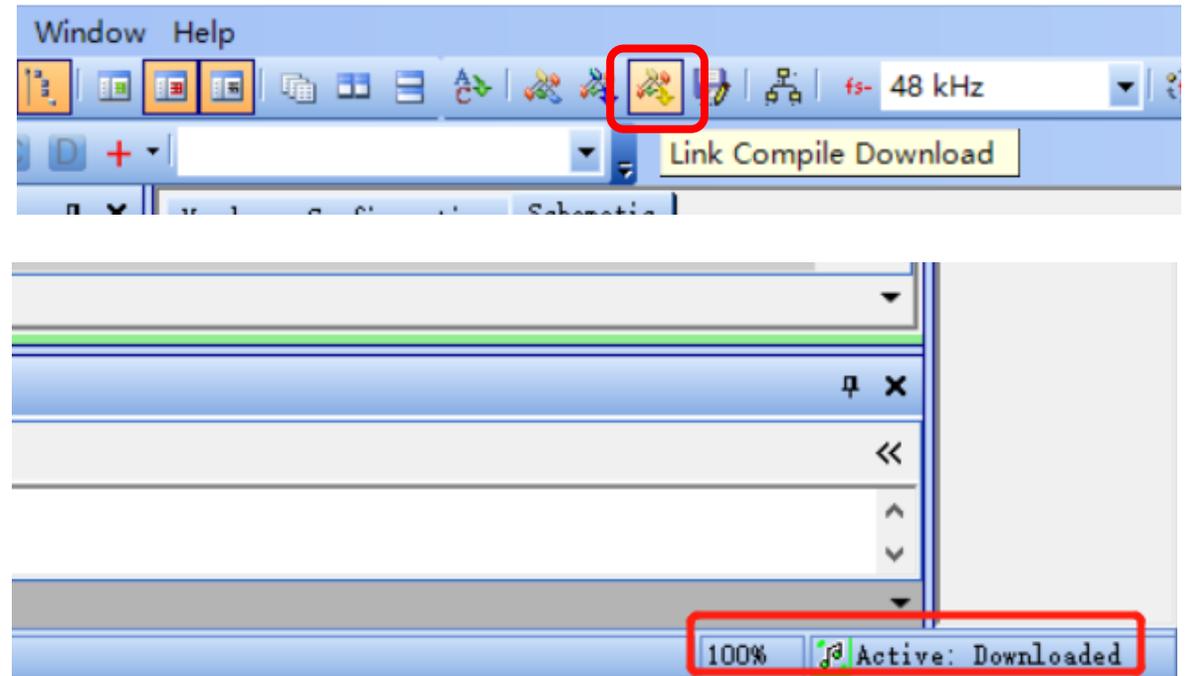
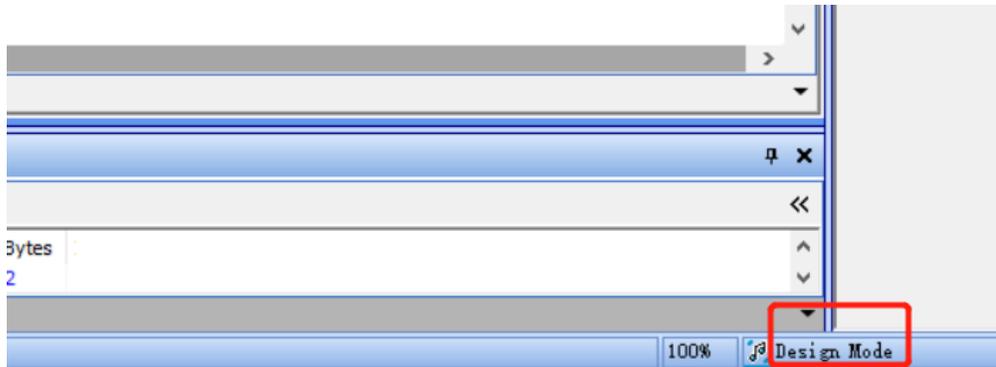
Register	Address	Value
Core	2076	b 0000000011100
GpioAll	2056	b 000000000000
RAM	2077	b 1000
SerialOut1	2078	b 00000000000000
SerialInput	2079	b 00000
MpCfg0	2080	b 000000000000111100000000
MpCfg1	2081	b 000000001111100000000000
AnalogPower	2082	b 000000000000
AnalogInterfa	2084	b 1000000000000000
AnalogInterfa	2085	b 0000000000000000



# Basic Program – Program Writing

How to run finished programs offline? We must write them into E2Prom.

- ① If it shows “Design Mode” at the bottom right corner, you need to click “Link Compile Download”. When it shows “Active: Downloaded”, we can move on to next step.





# Basic Program – Program Writing

- ② Click “Hardware Configuration”
- ③ Select “ADAU1701” module and then right click
- ④ Choose “Write Latest Compilation to E2PROM”, you will see the right window. Click “OK” and wait for finish.

