A²B MODULE AND EVM KIT USER MANUAL

For Revision 2 and 3 hardware

Rev 2.7

27-Sep-20

Signal Processing

http://clk.works/

Products covered:

AB0001 A²B to I²S module

AB0002 EVM for A²B module (Slave config)

AB0002M EVM for A²B module (Master config)

1. TABLE OF CONTENTS

1 INTRODUCTION

The Clockworks A²B module is intended to allow developers to quickly integrate Analog Device's A²B capability in to a prototyping environment. For OEMs it provides a quick time to market solution without need to design and debug the A²B portion.

The module can operate as either an A²B master or slave. In either case it is intended to be locally powered, i.e. it does not support drawing power from an A²B bus with phantom power. The module does however provide power to downstream A²B slave nodes using A²B's phantom power feature.

The module itself is a small 2 layer PCB with two 12 pin 2.54 mm (0.1") connectors on the bottom for easy mating to other hardware.

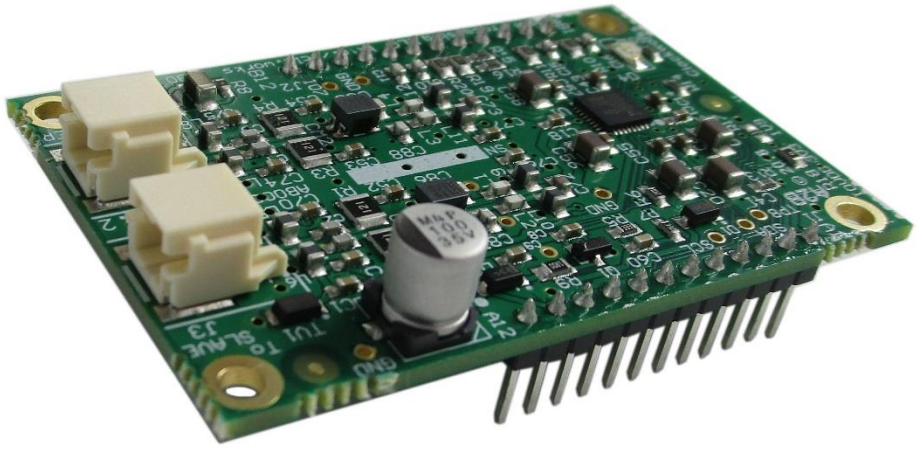


Figure 1 A²B module

1.1 MODULE EVM

For system developers there is a need to provide A²B slave devices that can stand in for other hardware, or to monitor audio streams that are being sent between other devices in the system. The EVM also provides a known good reference for operating the module and a design that can be copied if needed.

The EVM uses the Analog Device's (ADI) ADAU1761 Sigma DSP part, so the EVM can perform local processing on the audio input and output, which is a common practice in many A²B applications.

The standard EVM does not act as an A²B master. While Clockworks directly supports the ADI EVAL-AD2428WD1BZ and SHARC Audio Module as the A²B master for the provided examples, any A²B master can be used with EVM. Access to ADI's A²B tool set is needed to configure other masters to be used.

If your application requires an A²B master there is a separate orderable part number.

For master mode use an Analog Devices USBi (or equivalent) will be needed to use the SigmaStudio A²B development environment. The USBi is not needed for the slave mode EVM (but is probably needed for whatever master you are using, you just don't need a second one).

It is possible to convert a slave mode EVM in to a master mode EVM and vice versa, but it involves adding or removing parts as described in this manual.

1.1.1 IO7

The original EVM design supported prior generation A²B chips, and they required IO7 to be grounded during power on to place the A²B chip in slave mode. The newer parts, including the AD2428(W) used

on the Clockworks A²B module, rely on I²C commands to place the chip in Master mode, or upstream A²B activity to place the chip in slave mode.

Slave mode EVMs (rev 2) still have IO7 grounded, but there is a trace point where the copper can be cut.

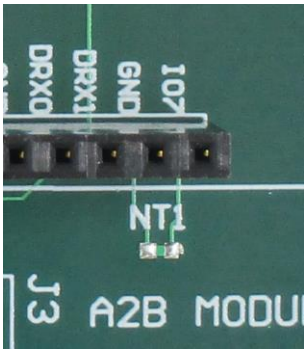


Figure 2 Cut NT1 to let IO7 float or be used for other things

On EVM Master mode boards ordered from Clockworks this trace is already cut and the pin is left floating.

1.2 SOFTWARE SUPPORT

At present all A²B software is supplied by ADI directly. Normally this software package includes both an add on for Sigma Studio that allows an A²B network to be described graphically, and a library with a standard API that can be used by the (host) processor that is connected to the A²B master (first) device.

The A²B API information can be found in the ADI document *"AD2421/AD2422/AD2425 Automotive Audio Bus A²B Transceiver Programming Reference"* document 82-100128-01, Rev 1.1 or as updated to the latest version. This guide, along with the A2B transceiver manual, is also needed to understand the register settings that are exposed in the A²B add-on for Sigma Studio.

Functionally the Clockworks module and EVM combination in slave mode operate the same as the ADI WBZ eval board, except that the Clockworks system is locally powered.[1]

Clockworks supplies SigmaStudio examples for the ADI SHARC Audio Module that allows audio to be routed between the EVM and the SHARC host.

When using ADI's tools for A²B configuration and operation remember to set the A²B device type for the module (EVM) to AD2428W.

Please see section 6 for an example with the A²B add-on for Sigma Studio.

[1] The Clockworks EVM setup matches the original ADI 2425WBZ design. The newer ADI 2428WBZ design adds two microphones and has a few other minor I/O changes. From the perspective of configuring the hardware in the ADI Sigma Studio for A²B tool they are all very similar.

2 GETTING STARTED WITH THE EVM

The setup steps are pretty similar between the two version (master or slave) of the EVM board.

The EVM board and the modules are ESD sensitive and must be handled following good ESD practices with a proper and safe grounding of the work area and anyone handling the boards.

The EVM kit ships with a 9V DC power supply that has universal input. The North America version uses the two blade plug, where as the international version of the kit has multiple adapters that mate with the supply.

If using the international version of the kit please attach the correct AC mains adapter to the supply.

The AC adapter supplies 500 mA at 9 VDC (center positive) via standard 2.1mm barrel connector. Alternately the board can be power from a bench supply providing 5 – 9V DC by connecting wires to the Phoenix block connector.

See Figure 2 and Figure 3 for the location on the hardware for the items discussed in the next section.

2.1 CONNECTION STEPS- SLAVE EVM CONFIGUATION

The EVM is shipped with an A²B module installed. The A²B module is configured for IO_{Vdd} of 3.3V

- Set SW1 (power) to the off position (slide button at board edge).
- Attach the 9 VDC supply to input jack J1 and plug the supply in to the AC mains.
    - **DO NOT CONNECT A 12V SUPPLY – YOU WILL RELEASE THE SPECIAL SMOKE**!
- Connect an audio source to J8 (Line in) if planning on using the EVM for stereo audio acquisition
- Connect headphones or amplified speakers to the Headphone out jack J4. Alternately amplified speakers or other line level device can be connected to J5 (Line out) if you change the default software for the ADAU1761.
- Connect an A²B cable to lower module jack ("From Master") to the A2B master or A2B slave located electrically closer to the A2B master.

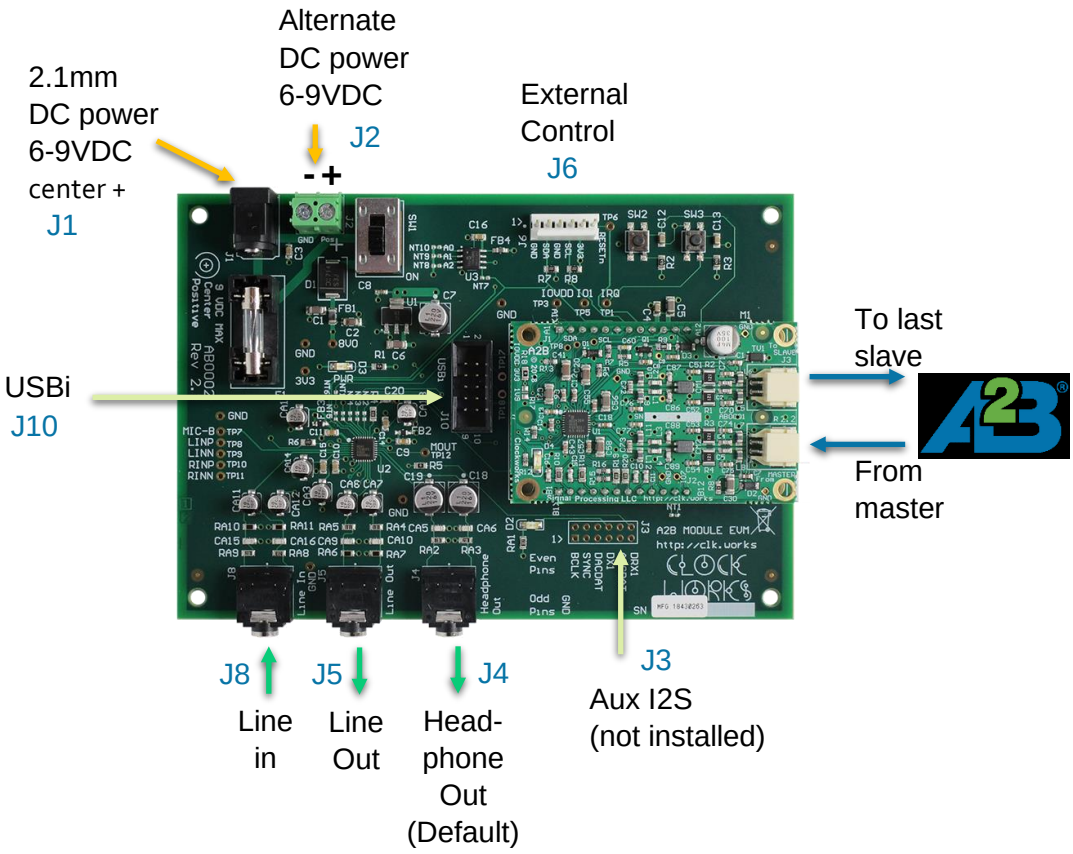


Figure 3 EVM external connectors

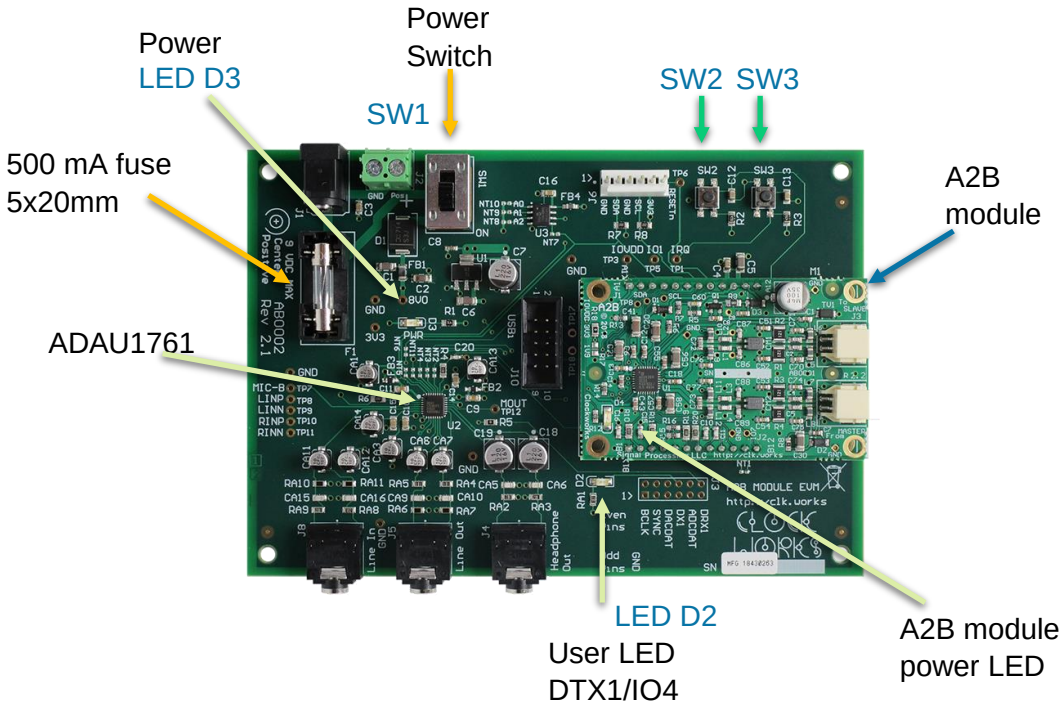


Figure 4 EVM Components

At this point the EVM can be powered up by switching SW1 to the "ON" position. The power indicator on the EVM and on the A²B module should be light.

Once the A²B host is operational and the A²B network established – which with the default software also includes programming the ADAU1761, the User LED (D2) will light.

If the user LED fails to light or the board is not operational for passing audio check that the download on initialization check box is enabled if using the Sigma Studio A²B tools. If developing application programmatically verify the ADAU1761 code is being correctly processed and downloaded.

As with the ADI A²B EVM boards this board includes an EEPROM. By default that EEPROM is not programmed. Some A²B software example may expect certain contents in the EEPROM.

2.2 CONNECTION STEPS – MASTER EVM CONFIGURATION

The steps are basically the same as the hardware is almost the same. The EVM that's been built to support master mode will have a MASTER sticker on it. The software settings are different, so please read the software section to make sure you understand the differences in the way the boards are configured.

We'll illustrate the connections using one Master EVM and one Slave EVM board.

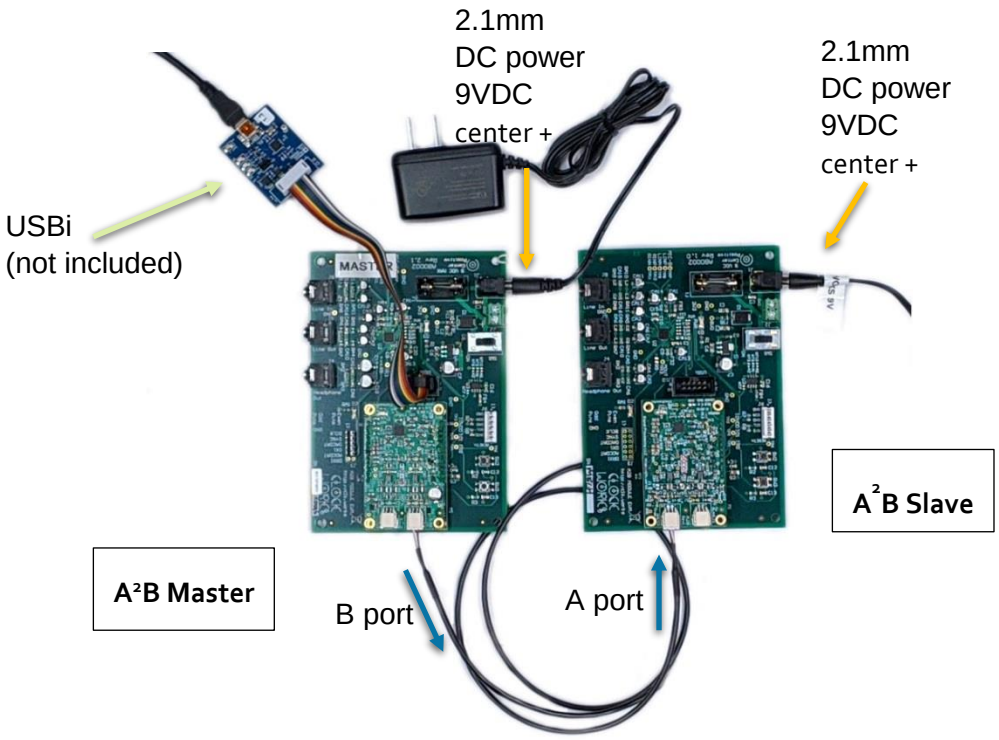


Figure 5 Setting up a master mode EVM

Follow the directions for the slave mode board as described in the prior section. Then plug in the USBi. A shrouded header is used and the USBi has a keyed 10 pin IDC connector.

An A²B cable is used to connect the master mode EVM's downstream port (B) to the upstream port (A) on the slave node EMV. If you have more nodes to connect they would connect to the downstream (B) port of the slave node.

Once the hardware is connected you can power up the boards, the order that they are powered up in does not matter. Please see the software section for information about configuring the boards.

2.3 PUSHBUTTON SWITCHES

The EVM includes two pushbutton switches that connect to the A²B module. Specific software must be used to define interactions with the switches. A small capacitor is used to aid in the debounce of the switch but software should verify a clean switch closure and release.

SW2 is connected to the DRX1 signal, which is also GPIO 6 on the AD2428 device.

SW3 is connected to the IRQ pin, which can also be used as GPIO 0 on the AD2428 device.

2.4 USE

There are two ways to evaluate A²B bus operation. One is with the Analog Devices A²B add-on for Sigma Studio with a supported ADI A²B bus master and the other is using the ADI developed API on your host processor.

Please see section 6 for an example with the A2B add-on for Sigma Studio. Guidance for direct software development is included with the ADI API documentation.

2.4.1 ANALOG OUT

The Analog Devices default DSP program uses the headphone output jack. To use the line out jack the default program must be modified to route in the I2S input to the DAC and enable both the headphone and line out jack in the chip register setup.

220uF capacitors are used on the headphone output for AC coupling. With 32 ohm headphones Fc is 22 Hz; 16 ohm headphones Fc is 45 Hz.[2]

For the line out 10 uF capacitors are used, with typical 10 kohm input impedances of an amplifier Fc is 1.5 Hz.

3 EVM DESIGN INFORMATION

When the EVM is purchased a design package is provided by email or an emailed link to download the information.

The package includes the following information:

- PDF of schematic. Part information is embedded in the PDF, click on a part symbol to see detailed information on the part.
- 3D PDF of the board. Note that Acrobat Reader security will ask if you trust the document. You must click yes. You then may need to click in the blank field to make it display. Some versions of Reader have a bug and will not display unless the document is saved locally. You can enable display of various board features and components using the standard 3D Acrobat features.
- Zip with Gerbers.
- Altium project files
- BOM as an Excel spreadsheet
- .STEP file for the A²B module

[2] So yes, if you have low impedance headphones and want a good experience you probably want an external buffer amplifier if you wanted to do some sort of critical listening.

3.1 POWER INPUT

The EVM design include a diode on the DC input to prevent damage from a reverse connected input supply. This diode also reduces the input voltage by about 0.7V; on the board/schematic this is rounded off to call the supply an "8V" supply; the actual voltage is about 8.3V.

The AD2428 maximum supply voltage is 9V and this small extra drop prevents operating the AD2428 on the module near it's limit. The module design includes a Schottky diode as suggested by ADI's design guide for a local powered A2B module, meaning there is another 0.4V or so drop on the A²B module and this the actual voltage to the AD2428 on the module is about 8V.

The only possible impact from the lower supply voltage is if A²B phantom power feature is used to power multiple downstream slave nodes; each node has a 0.4V drop. For more details please see the full AD2428 datasheet for a discussion on phantom power of slave nodes.

The EVM board design includes a fuse. Under normal use the fuse will never blow as the A²B is protected from catastrophic failure for a range of open and short to power conditions. While developing a system it is certainly possible to make an error (for example modifying the module in some way) that would overload the supply, so the fuse serves as a little protection against doing too much damage. Looked at this way, if the fuse blows you made a big oops and it will be time to buy a new board.

The PCB is a four layer board, the internal planes carry GND and +3.3V. The power input and 8V power are carried by traces.

The EVM does not use the IOVDD module output. The A²B module must be jumpered for 3.3V IO when used with the EVM.

3.2 FEATURES FOR MODIFYING THE EVM

There may be cases where a change to the default wiring is needed. There are a couple of provisions in the design to attempt to simplify making modifications. Please note that Clockworks can not support modified boards nor take any responsibility for your modifications making you late for dinner or something worse like burning down the building.

3.2.1 TESTPOINTS

Testpoints are available on some of the signals to facilitate probing or modifying the board. Please see the schematic for their (electrical) location.

The unused analog I/O pins of the ADAU1761 are also brought out to a test point; please see the ADAU1761 datasheet for how to use them as no circuitry is provided on the EVM.

In addition to signals a few GND test points are provided. These are connected to the internal ground plane of the PCB. There is a single ground plane in this design.

3.2.2 NT COMPONENTS

NT (Net Tie) components are 0603 footprints with copper trace between them. By cutting the traces the two net pieces can be isolated and connected to different places. They are available on the ADAU1761's I²S signals and the 3 address bits for the EEPROM.

NT7, located on the SDA pin of the EEPROM, would normally not be used for any purpose.

NT1, connected to the IO7 pin of the module, must be grounded to have the AD2428 operate in A²B slave mode. If the trace is cut and the pin pulled high then the module will operate as a master and you must provide an external host to control the AD2428 over I²C. If this is changed the I²S clocking must also be changed to match the needs of the AD2428 and the programming of it in master mode.

3.2.3 MICROPHONE INPUT

The EVM provides test point holes for the microphone pins of the ADAU1761. Please see the ADAU1761 datasheet for suggested circuit configurations for using these inputs.

Figure 6 ADAU1761 microphone input signal breakout

3.3 J6 EXTERNAL CONTROL CONNECTOR

This 6 pin connector is a TE MTA-100 series part, 640456-6. It follows the same pinout as used on some other Clockworks products. Pin assignments are as follows:

1	GND
2	SDA
3	GND
4	SCL
5	3.3V
6	RESETn

The RESETn is normally defined as an active low signal that is driven by open collector logic where the default driving source includes a pullup. On the EVM board and the A2B module there is no reset capability, the parts have built in brown-out and reset. Therefor this signal serves no purpose for the normal EVM use.

If using the I²C bus keep in mind that the AD2428 device on the A²B module is normally the I²C master.

3.4 J3 I²S CONNECTOR

This 12 pin dual row connector (2.54mm/0.1" spacing) provides access to the I²S signals. It is not installed on the EVM so users can insert whatever connector may be needed.

2	BCLK
4	SYNC
6	DACDATA (DTX0)
8	DTX1
10	ADCDAT (DRX1)
12	DRX1 (IO6 - SW2)
1,.., 11	GND (odd pins)

4 USE OF THE EVM WITHOUT AN A²B MODULE

The EVM Board can be used without an A²B module for ADAU1761 development using ADI's Sigma Studio tools. The ADAU1761 receives its clock on the MCLK pin, it will not operate without that signal. The default software expects a 12.288 MHz clock which corresponds to 256 Fs for an Fs of 48 kHz.

There are two ways to provide this signal. One is to connect a 3.3V clock input to pin A3 of the module connector (CLK on the schematic).

The other is to install a 12.288 MHz oscillator at X01 (Kyocera 3225 series or similar) along with R9 (33 ohm 0805 footprint) and C17 (0.1 uF 0805 footprint). If those parts are installed a module in slave mode can not be used as the module must be the device supplying the I²S clocks for an A²B slave device. This would, with an external control processor or the USBi, allows the EVM to be the A²B master.

The EVM master mode board comes with this oscillator installed, therefor the A2B module most only be run as a master (I2S slave) to prevent signal conflicts

Other oscillator frequencies can be used, the ADAU1761 datasheet provides more detail. Using the suggested 12.288 MHz value keeps the settings the same between an A²B application and non A²B application use.

When ordered from the factory as the Master mode version of the EVM, a 12.288 MHz oscillator is used for all current hardware revisions (older revisions may have been supplied with 24.576 MHz oscillators). The choice of MCLK will affect the ADAU1761 setup, MCLK is not used by the A²B module.

4.1 ADD COMPONENTS

Three SMD components, an oscillator, resistor (series terminating so typically 33 ohms) and a decoupling cap can be solder to the EVM board as a permanent change. This may be desirable if hacking the EVM to operate as an A2B master (which requires an external processor to act as the controller).

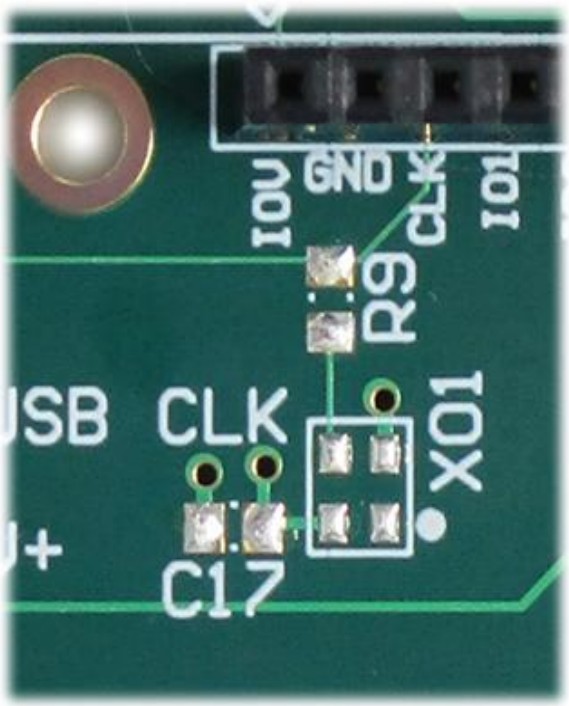


Figure 7 Location of optional components on EVM for local clocking (installed on EVM master version)

If the module is operated in master mode (see notes regarding use of NT7) then this oscillator would provide the master clock. The operation of the bit clock and frame syncs must be decided and the ADAU1761 and AD2428 programming matched with the specifics of the I²S signals in this scenario.

4.2 TEMPORARILY WIRE IN AN OSCILLATOR FOR USE WITHOUT A²B

The control header (J6) can be used to supply power to an oscillator who's output is plugged in to the A²B header as a way to use the EVM as a SigmaDSP development environment without making a permanent change to the board.

Figure 8 shows two examples of how an oscillator (12.288 MHz) can be connected. J6 on the EVM is used to supply power and the clock output is wired to a pin. The mating connector is an MTA-100 series part, TE part number 3-640440-6 is for 22 AWG wire.

Heat shrink tubing helps maintain mechanical robustness and offers some protection against accidental shorts.

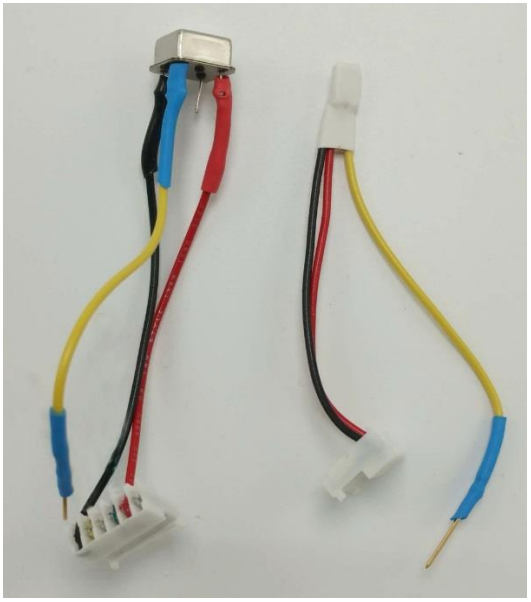


Figure 8 Oscillator options example for using EVM without A²B. Can style on left, SMD style on right (under shrink wrap).

The clock output (yellow wire) is connected to pin A2 of the header.

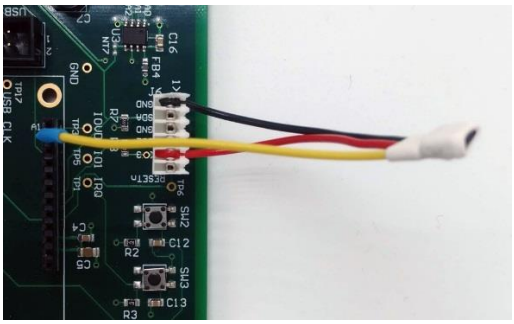


Figure 9 Oscillator connection example

4.3 USBI CONNECTION

For use in master mode, or to develop software for the ADAU1761 Analog Device's Sigma Studio tools are used. An emulator (ADI calls it *USBi*) is required for downloading and interacting with the ADAU1761. For more information please see:

http://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/eval-adusb2ebz.html#eb-documentation

5 MODULE INFORMATION

All module functions are determined by the AD2428W device, please refer to that device's datasheet and technical reference manual for detailed operating information.

5.1 CONNECTOR

Pinout – A Connector, J1 on PCB label.

Pin	Name	Notes
1	IOVDD	Jumper selects between AD2428 internal regulator voltages, default is 3.3V (vs. 1.8V)
2	GND	
3	ADR2	AD2428 ADR2/IO2 line with 10K pulldown
4	ADR1	AD2428 ADR1/IO1 line with 10K pulldown
5	IRQ	AD2428 IRQ/IO0 line
6	GND	
7	SDA	I2C data
8	SCL	I2C clock
9	GND	
10,11	Vxx	Power 5-8V suggested range, 4V min, 9V max.
12	GND	

Pinout – B Connector, J2 on PCB label.

Pin	Name	Notes
1	GND	
2	BCLK	AD2428 bit clock (output from module as a slave)
3	GND	
4	SYNC	AD2428 frame sync (output from module as a slave)
5	GND	
6	DTX0	AD2428 DTX0 with 10K pulldown (output)
7	DTX1	AD2428 DTX1 with 10K pulldown (output)
8	GND	
9	DRX0	AD2428 DRX0
10	DRX1	AD2428 DRX1
11	GND	
12	IO7	AD2428 IO7/PDMCLK pin[3]

[3] Original rev 1 modules were built with AD2425W and this pin needed to be grounded to indicate a slave device. There is a jumper on the EVM that can be cut to remove the ground connection on this pin if needed (the EVM is however only capable of being an A²B slave)

5.2 MODULE MECHANICAL INFORMATION

Top View.

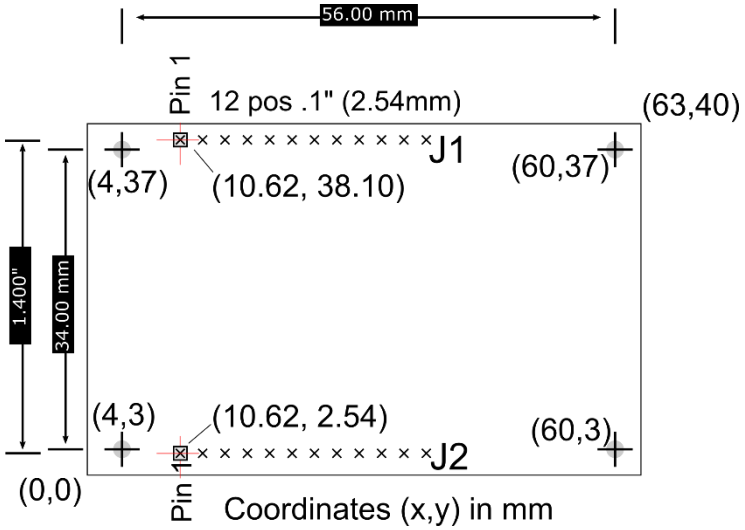


Figure 10 A²B module dimensions

The board is slightly asymmetrical relative to the mounting holes, which are 4mm from the back edge and 3mm from the front (A²B connector) edge.

All electrical components are on the top side of the board. The two 12 pin connectors are located on the bottom side of the board.

Component heights are nominally as follows for the figure shown below. Component substitutions during assembly may sometimes be required; this may alter these values.

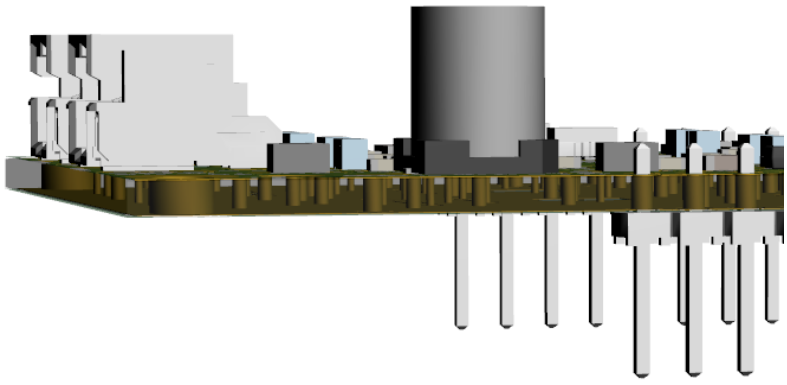


Figure 11 A²B module side view cut-away

Board thickness 1.6 mm(.062").

Plastic on bottom connectors thickness 2.29mm (.090").

Pin length 6.1mm (.24").

DuraClik connector height 6.4mm (.252").

C24 (bulk 100 uF cap) height 7.7mm (.303")

5.3 INTERFACE DETAIL

The following two schematic sections from the A²B module are provided to aid in understanding the A²B module interface.

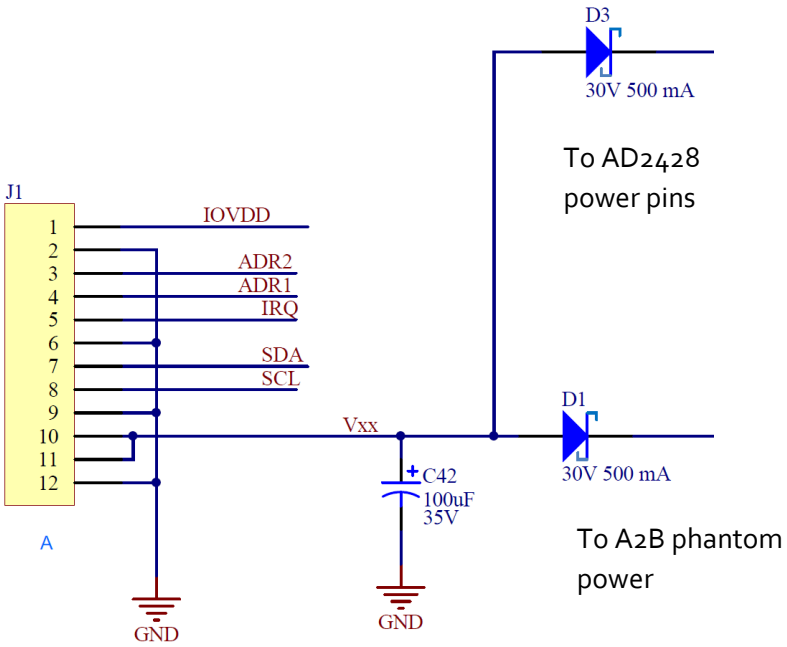


Figure 12 Module A connector wiring[4]

C42 is required by the ADI design rules and should be kept in mind for the load capacitance on the external supply.

The other connector carries the I²S signals. Analog Devices suggests series termination and a small 27

[4] RefDes for Rev 2 module PCB shown unless noted otherwise.

pF capacitor to reduce EMI issues and the module implements those suggestions.

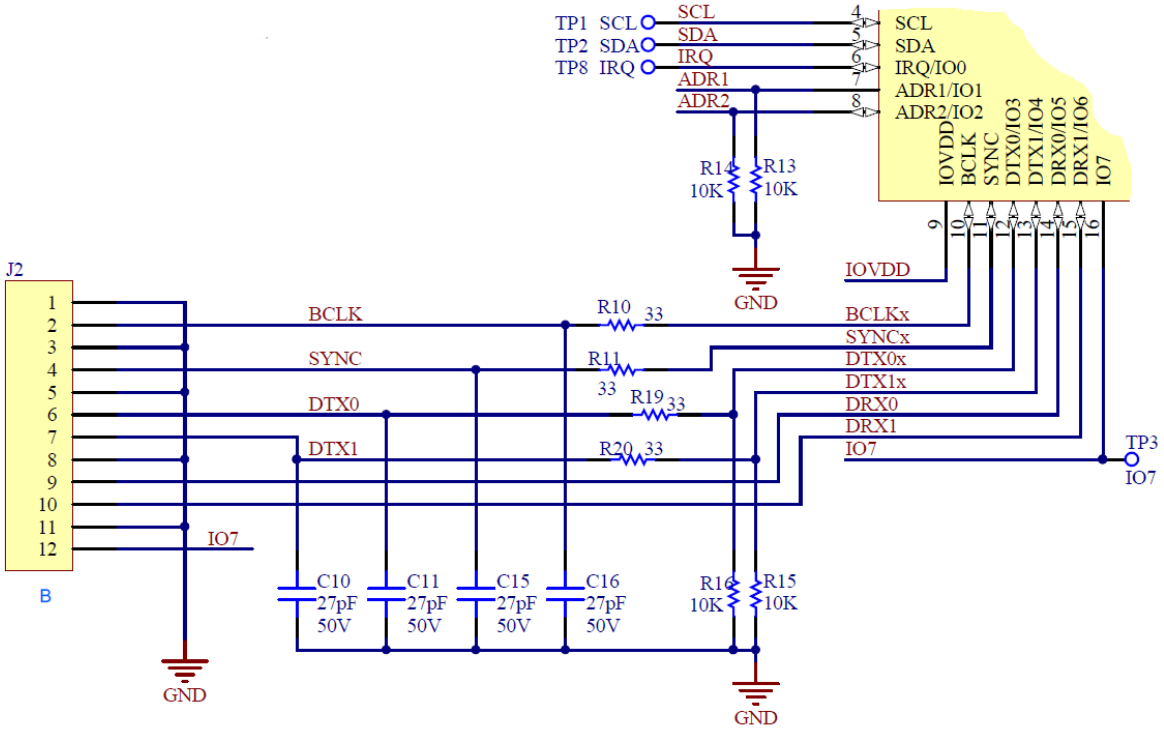


Figure 13 Module B connector wiring. On AD2428W IO7 can also be PDMCLK out pin.[5]

The remainder of the module schematic corresponds to the ADI recommended design for locally powered nodes, as that guidance is subject to change please see the ADI documentation for further details.

Unused module pins may be left floating except for the I²C pins. (see 5.3.3).

5.3.1 IOVDD

This pin is normally an output from the module. It can be configured for 3.3V or 1.9V[6] operation.

If neither voltage selection jumper is installed then IOVdd can be set externally, however all of the AD2428 rules about supply sequencing must be strictly adhered to.

The I²C bus must match the IOVdd, i.e. the pullup resistor's connection to the supply.

[5] RefDes for Rev 2 module PCB shown unless noted otherwise.
[6] The datasheet for the AD2428W specified the range of voltage from that supply as 1.7 to 1.98 with 1.9 being the typical value. In a few places though it's referred to as the 1.8V supply.

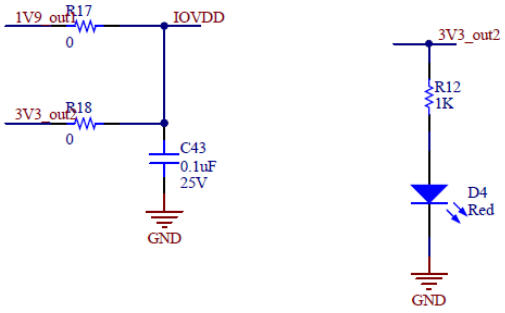


Figure 14 Rev 2 PCB IOVdd connections. 1V9 is connected to the AD2428W VOUT1, pin 32. 3V3 is connected to the AD2428W VOUT2, pin 29.

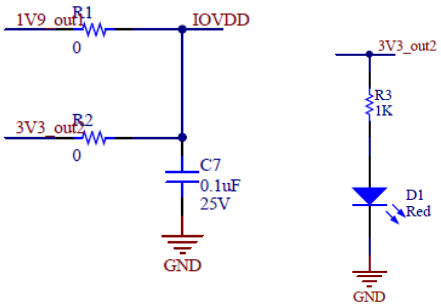


Figure 15 Rev 3 PCB IOVdd connections. 1V9 is connected to the AD2428W VOUT1, pin 32. 3V3 is connected to the AD2428W VOUT2, pin 29. (different RefDes #s)

Figure 16 Rev 2: IOVdd jumper resistors. 3.3V is the factory default

5.3.2 VXX

It is AD2428 datasheet shows that this pin should not exceed 9V, though the Schottky diode in the module input supply connection does mean that 9.3V is closer to the absolute maximum limit of the

module. The minimum voltage is 4.2V for the module to operate, however no phantom power can be supplied at that voltage.

5.3.3 SCL, SDA

As stated on the AD2428 datasheet, these two pins should be grounded if not used by system this board plugs in to.

If I²C is used then the main board must provide the I²C pullups and correctly match the IOVdd.

6 CONFIGURATION EXAMPLE WITH SIGMA STUDIO (EVM SLAVE MODE)

This section outlines the steps for using the Clockworks EVM with the Analog Devices A²B add-on for Sigma Studio. ADI includes tutorial examples with the software add-on and this section shows how to use the example ..\Schematics\BF\A2BSchematics\adi_a2b_3NodeSampleDemoConfig.dspproj

You can also create a new example on a blank sheet Sigma Studio sheet following the procedures in the ADI documentation. When first starting out it may be easier to edit a provided example for your host (and save it as a new file!) than to get all of the settings in place.

This section assumes the use of one of the standard ADI A²B EVM boards as the A²B master. If using the SC589 based SHARC Audio Module please also see the next section as that board is currently not covered by the ADI A²B package. While the concepts are the same as working with the ADI EVMs there are a few differences that must be considered.

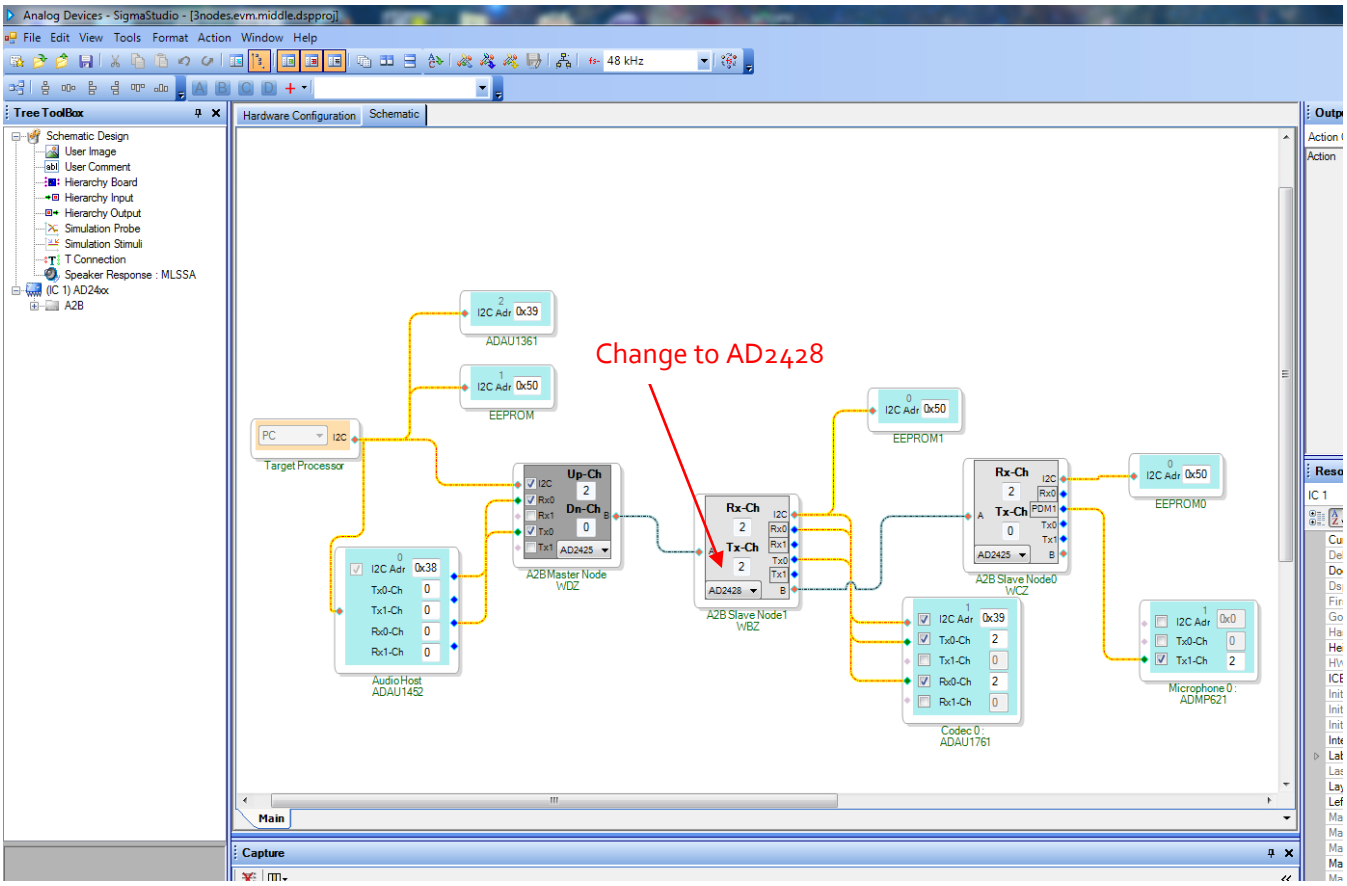


Figure 17 Sigma Studio 3 node example

Figure 6 shows one of the ADI default example with the (ADI WDZ board) ADAU1452 master and WBZ and WCZ slaves. The Clockworks EVM replaces the WBZ slave in this example. To do that change the chip type on the 2nd node to AD2428 as shown by the red arrow.

If you don't have the WCZ microphone board you can delete it from the example.

In this example the master node must be programmed, which is enabled by selecting its properties and checking the *Program during discovery* box. The path to the XML file with theADAU1452 must be valid and if just trying the system for the first time should be the default pass through program.

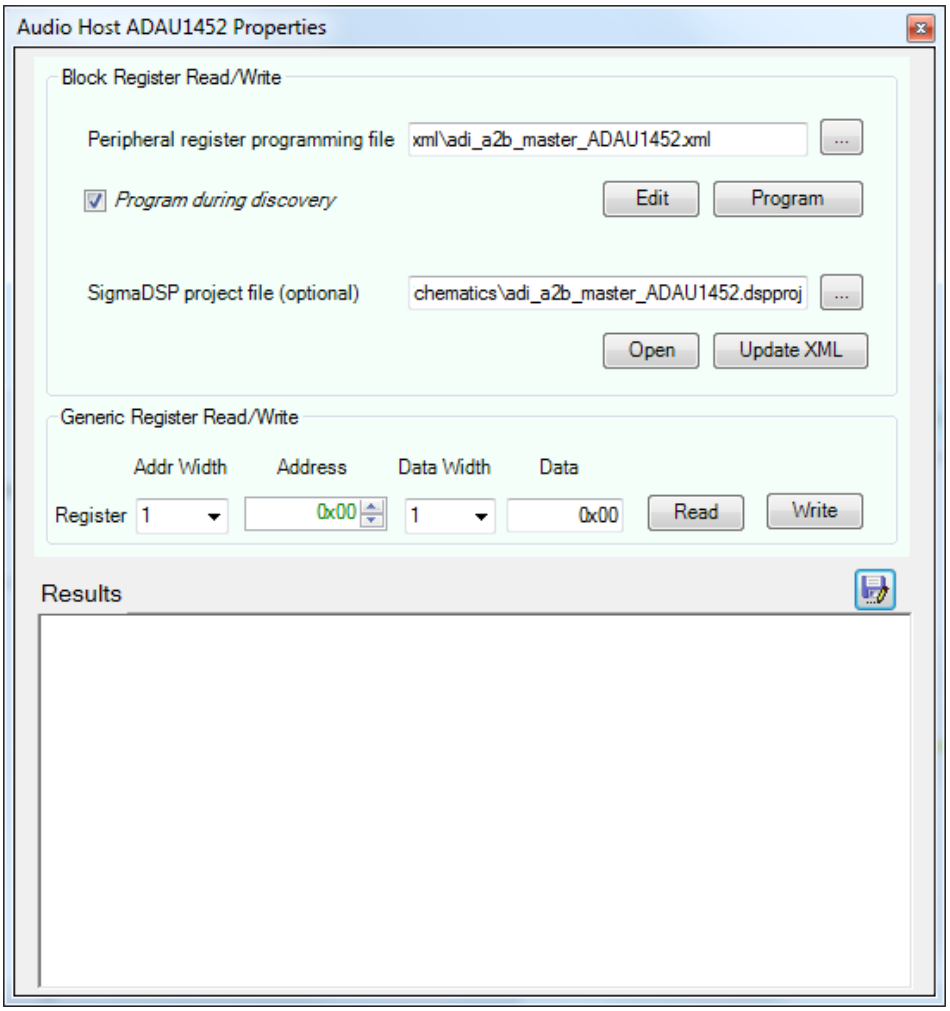


Figure 18 WDZ Master node properties

The Clockwork's EVM board ADAU1761 must likewise be programmed, and the default program used with the WBZ board can be used, as shown in Figure 8. As with the master node the *Program during discovery* box must be checked.

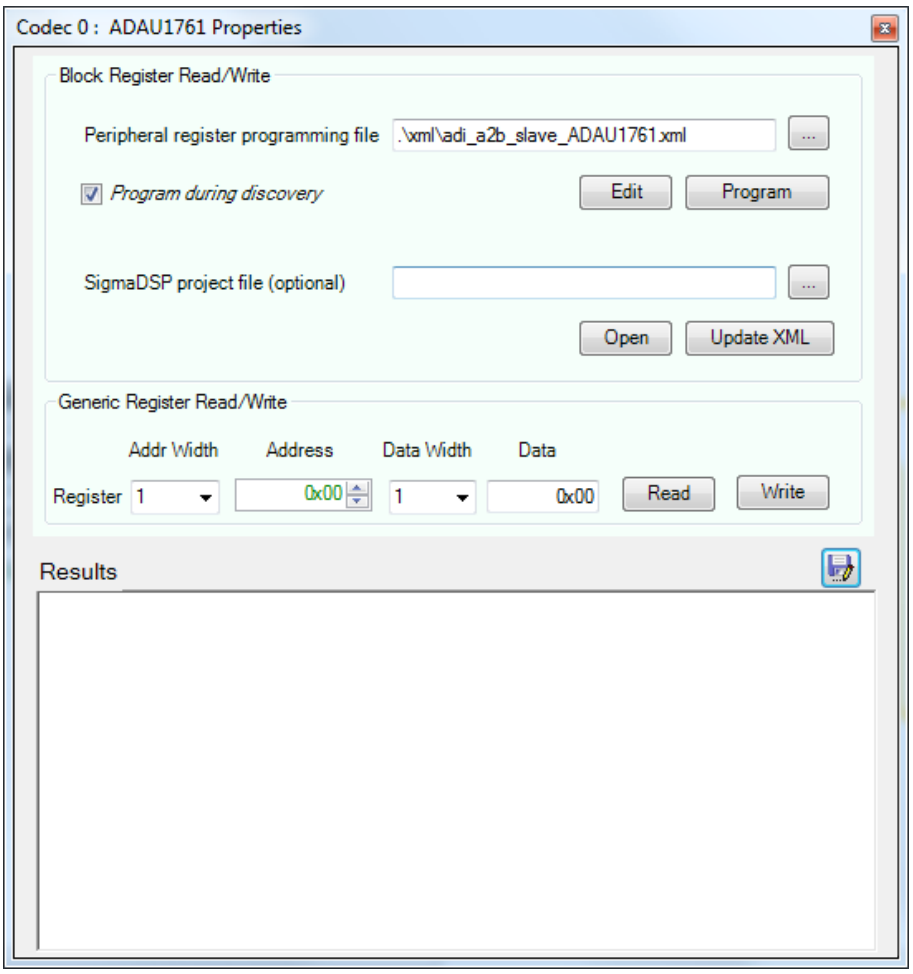


Figure 19 ADAU1761 programming on Clockworks EVM board (as a slave node)

It should not be necessary to change the AD2428 properties, the I²C setup screen (properties home page) is shown in Figure 9.

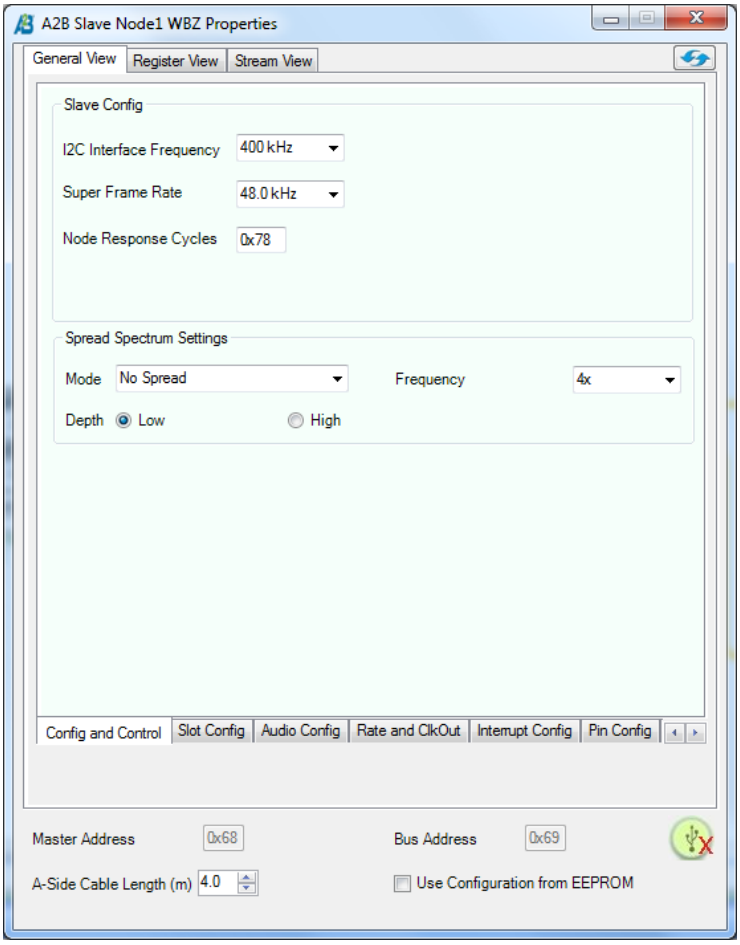
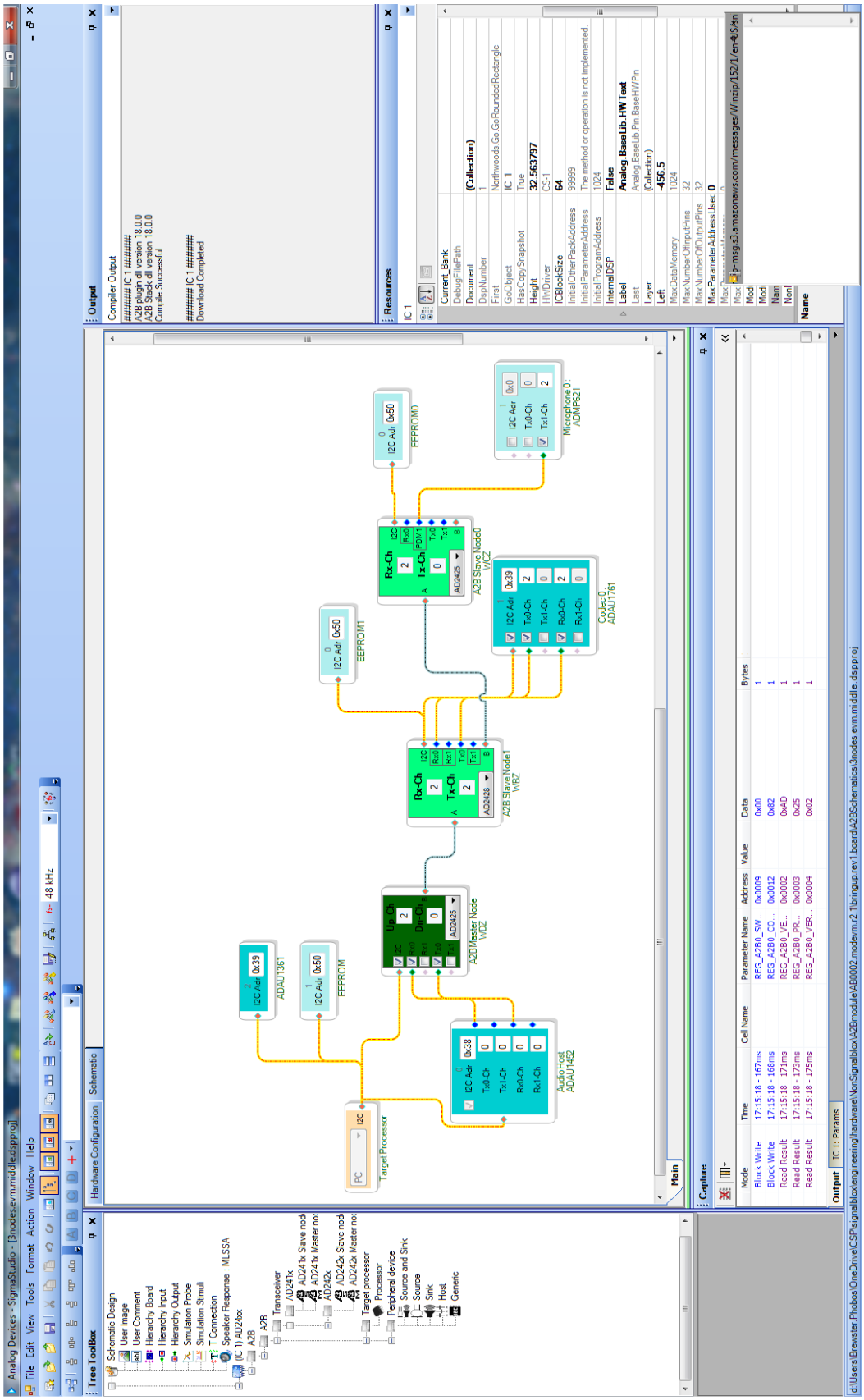


Figure 20 AD2428 Properties for Clockworks EVM based on defaults for ADI WBZ board.

Once all of the settings are checked hit the Link Compile Download button and after a few seconds the A²B nodes should turn green to indicate the networks has been discovered, as shown in the figure on the next page.



This particular ADI example takes the audio from the 2nd slave node (a WCZ mic board) and plays it out the headphone jack of the 1st slave node (the Clockworks EVM here, a WBZ board in the original example). The line in to the 1st slave node is sent to the headphone out of the A2B master.

6.1.1 I²S SETUP

If using the ADI provided example all of the settings are correct for the AD2428 to exchange audio with the ADAU1761 when it runs the corresponding default program. If inserting a A²B transceiver block and peripheral block (in stead of copying) all of the settings need to match that used for the SigmaDSP's I²S interface. Failure modes include no audio, very badly distorted audio, or full scale modulated noise.

Figure 21 and Figure 22 show the two relevant settings for use with the default ADAU1761 program.

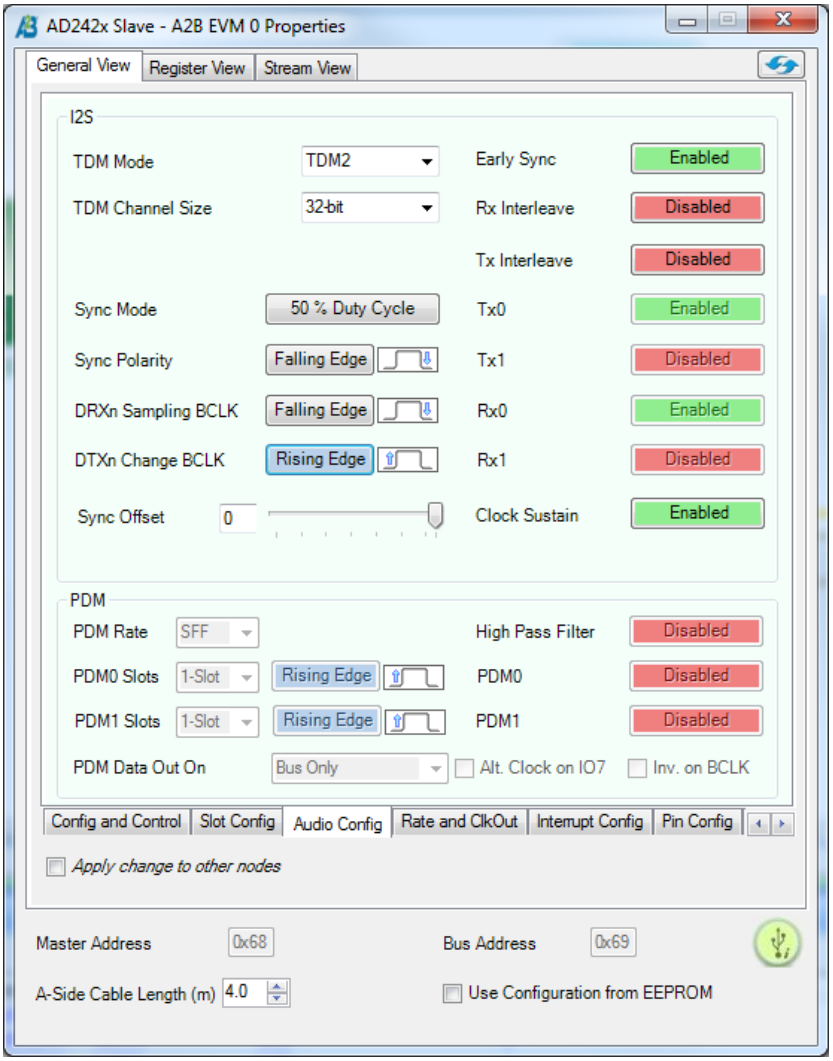


Figure 21 Audio Config tab settings for ADAU1761 default EVM program

If any of the edge or Sync setting are wrong highly distorted audio will result. These settings correspond in part with the `I2SCFG` register (0x42), which should have a value of 0x91 if the bits have been set properly.

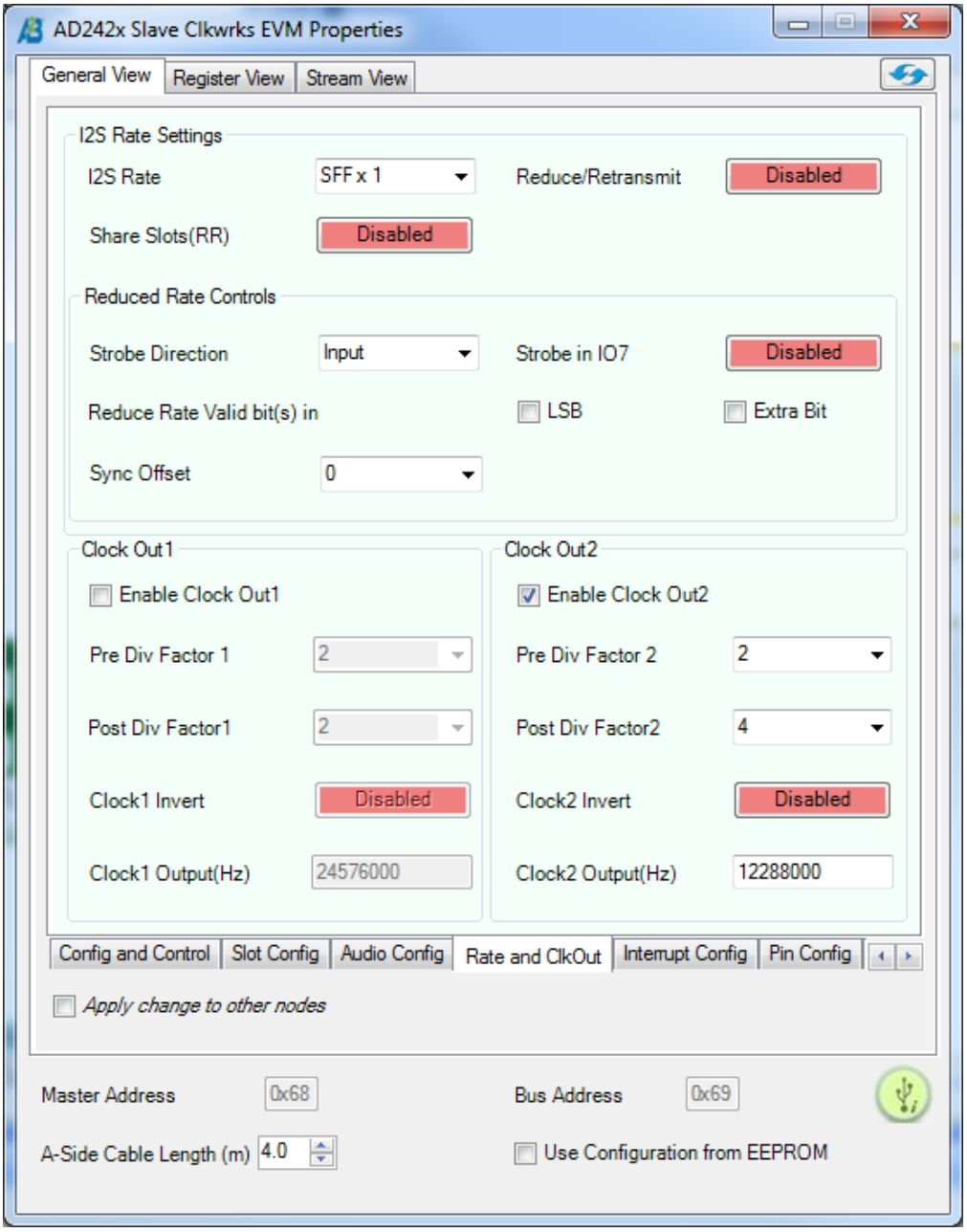


Figure 22 Rate and ClkOut tab settings for ADAU1761 default EVM program (slave node)

6.2 ACCESSING THE LED AND SWITCHES FROM SIGMA STUDIO

GPIO can be accessed from the AD2428 block in Sigma Studio by right clicking on the block associated with the EVM board. There are two ways to do this, one at a low level of direct register access, like what a program would do, and the other via the more convenient "General View," though that second method only helps for output, i.e. turning the LED on and off.

6.2.1 THE HARD WAY

Select Register View and the IO tab. For details of the registers and their bit maps please see the AD242x Programmers Reference Manual.

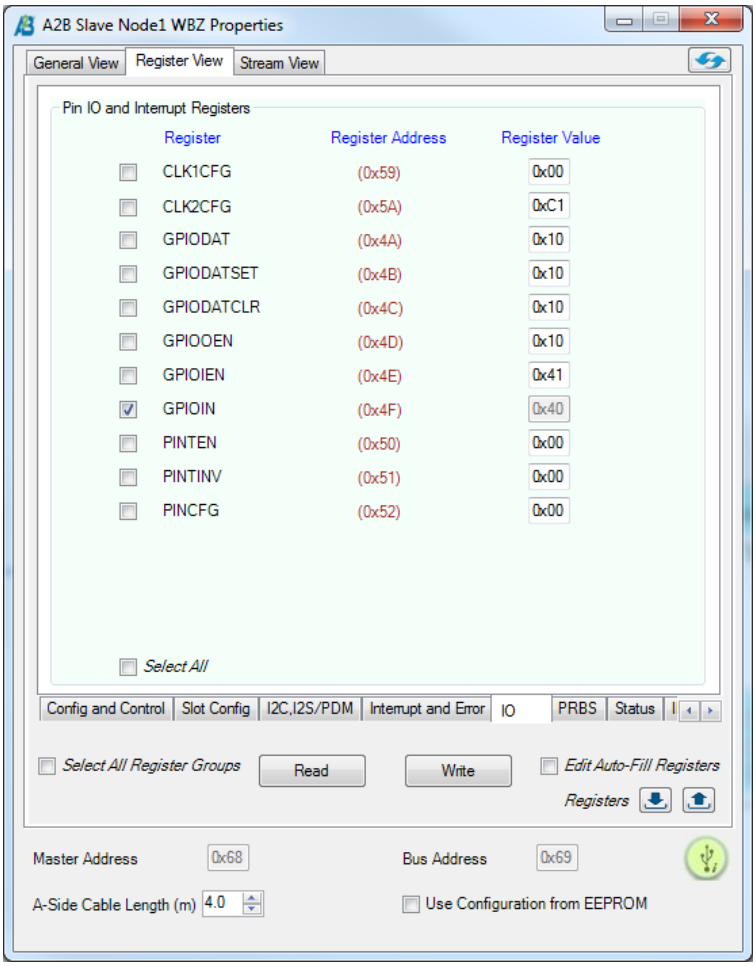


Figure 23 IO tab in SigmaStudio

Top operate the LED first write 0x10 to the GPIOOEN register (0x4D). Then place 0x10 in the GPIODATSET or GPIODATCLR register and write that to turn the LED on and off. The LED is connected (through the EVM to A2B module connector) to DTX1/IO4 pin of the AD2428.

To read the switches for write 0x41 to GPIOIEN register (0x4E). SW2 is connected to DRX1/IO6 (bit map 0x4)oand SW3 to IRQ/IO0 (bit map 0x01). Select GPIOIN and hit the Read button to see the switch values.

6.2.2 THE EASY WAY

This will allow turning the EVM's LED on and off, which might be useful to confirm the node you've selected is the one you think it is. On the General View tba select the Pin Config dialog and set GPIO4 to be an output. High turns the LED on, Low shuts it off.

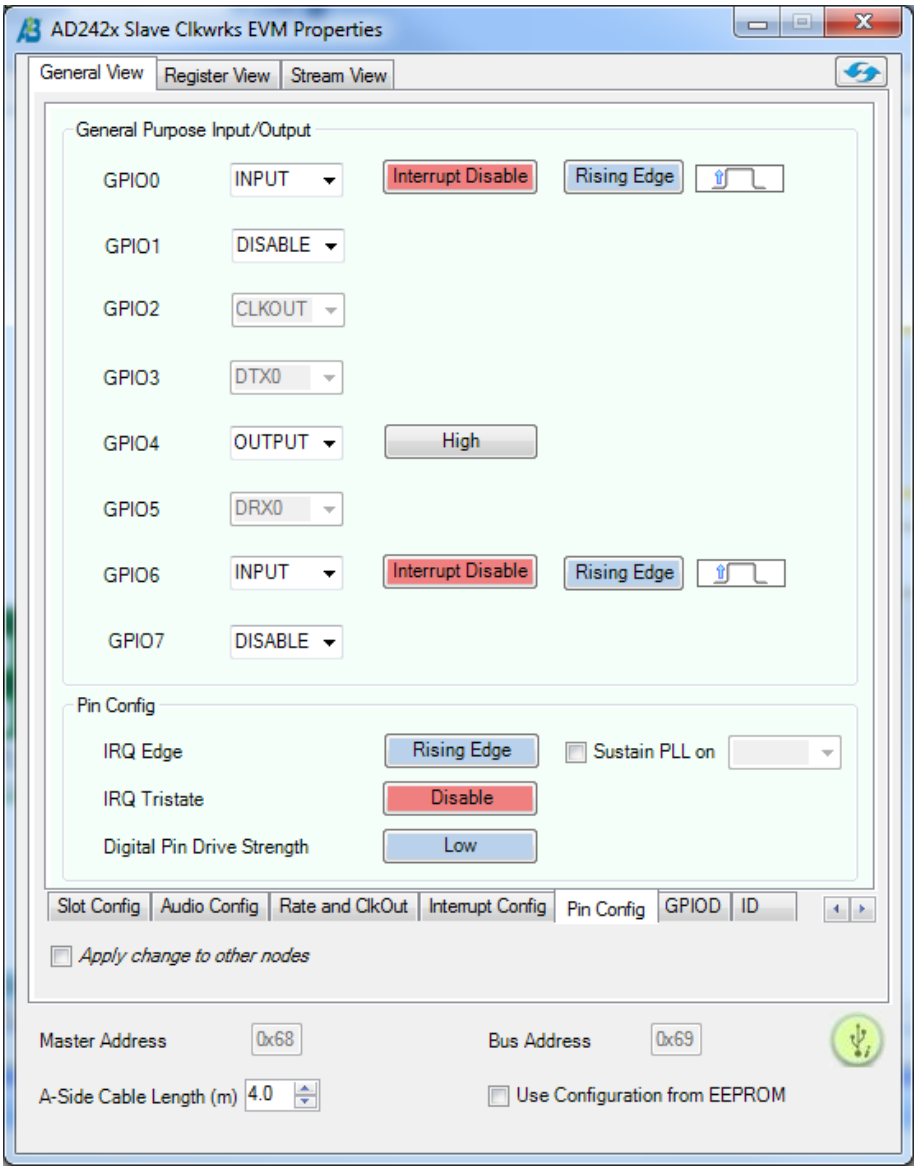


Figure 24 EVM User LED (GPIO 4) control

7 CONFIGURING A MASTER MODE EVM

Start with the supplied example (`EVM_mstr_EVM_slave_WCZ.dsproj`) or create a new project and add the host, AD2428, and ADAU1761 blocks as shown. If creating from scratch make sure you use the AD242x Master node for the first one.

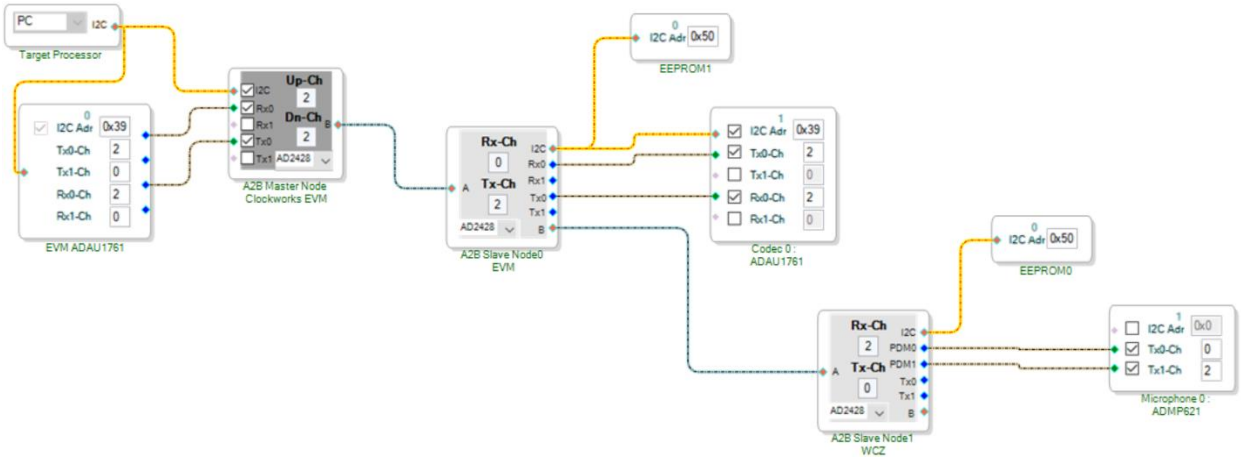


Figure 25 EVM master mode with EVM slave mode example (with WCZ node)

For the slave node follow the directions in the prior section, the remainder of this section will just look at the master node setup.

The AD2428W and the ADAU1761 are both controlled over I²C from the host running SigmaStudio. When a *Link Compile Download* is performed the AD2428W is initialized and the ADAU1761 program is also loaded as the *Program during discovery* is enabled in the ADAU1761 properties:

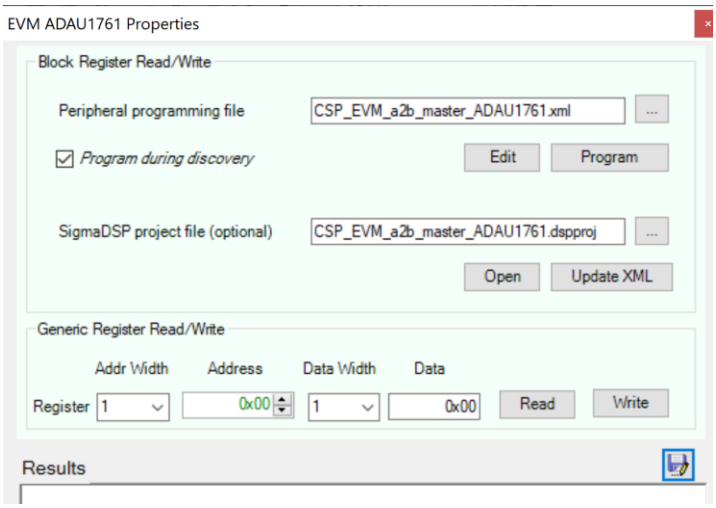


Figure 26 Master node 1761 file setup

The master node has slightly different properties than the slave node.

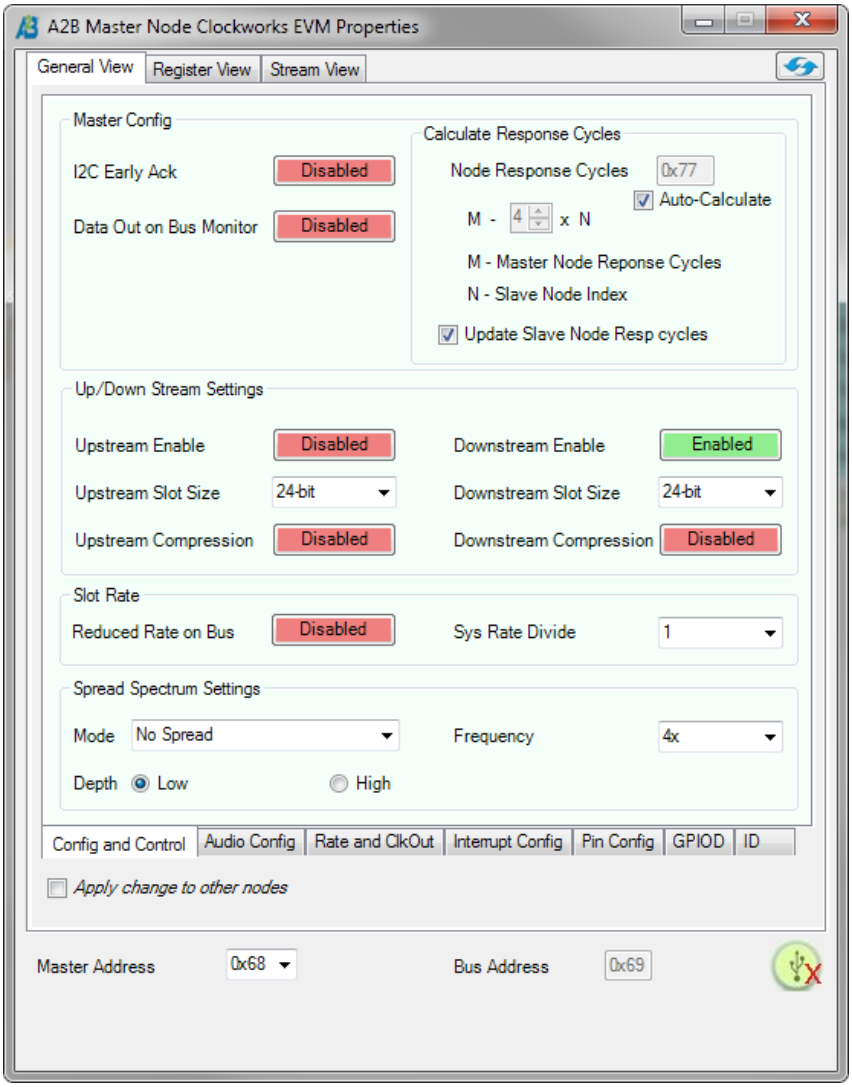


Figure 27 Master node properties

In the provided example the I²S lines between the AD2428 and the ADAU1761 are operated in TDM2 mode (more commonly called stereo). The ADAU1761 only has one I²S in and one I²S out line. On the EVM master node the ADAU1761 generates the I²S BLCK and SYNC signals; the opposite is true on an EVM slave node.

For the AD2428 this can be set directly in its properties:

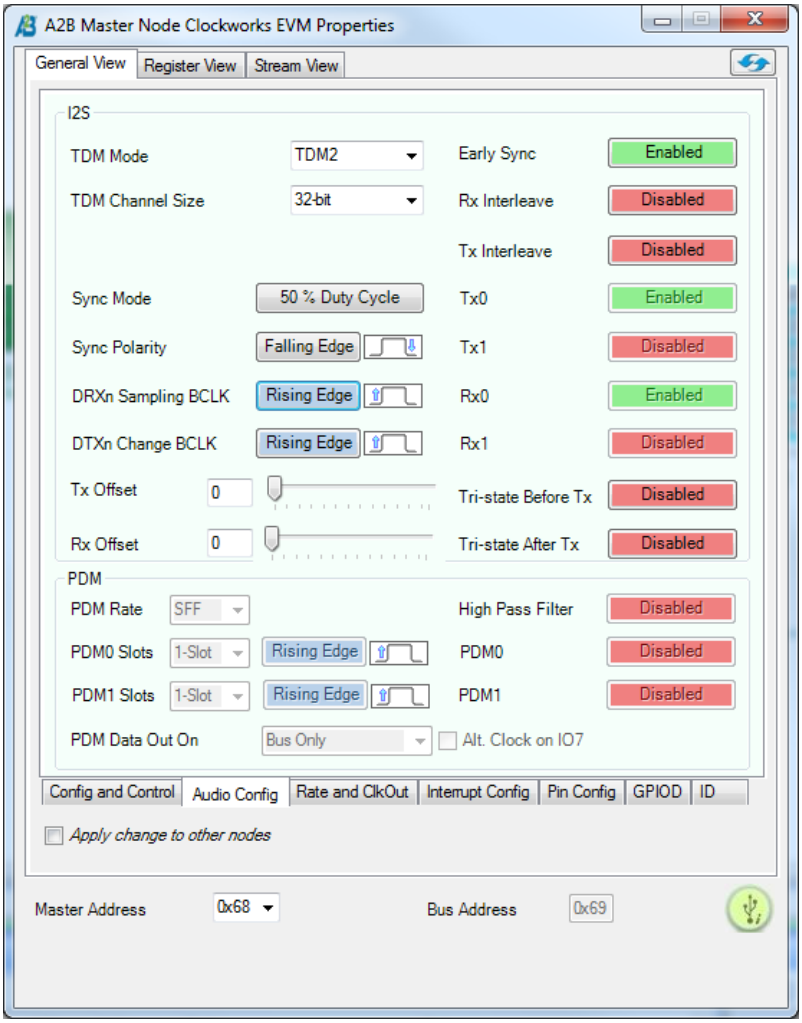


Figure 28 Master node settings for TDM2 over I²S

To set up the ADAU1761 the SigmaStudio schematic for a separate project most be updated. Do not mix up the master nodes ADAU1761 SigmaStudio schematic with the slave node's, the need to operate differently due to the different I²S clocking.

The provided master node ADAU1761 example is copied from one that ADI uses in their examples. Don't worry if the example(s) you have look different. You will want to edit these or create new ones to match your processing configuration.

The ability to create test tones is useful when debugging why nothing seems to be working the way you wanted.

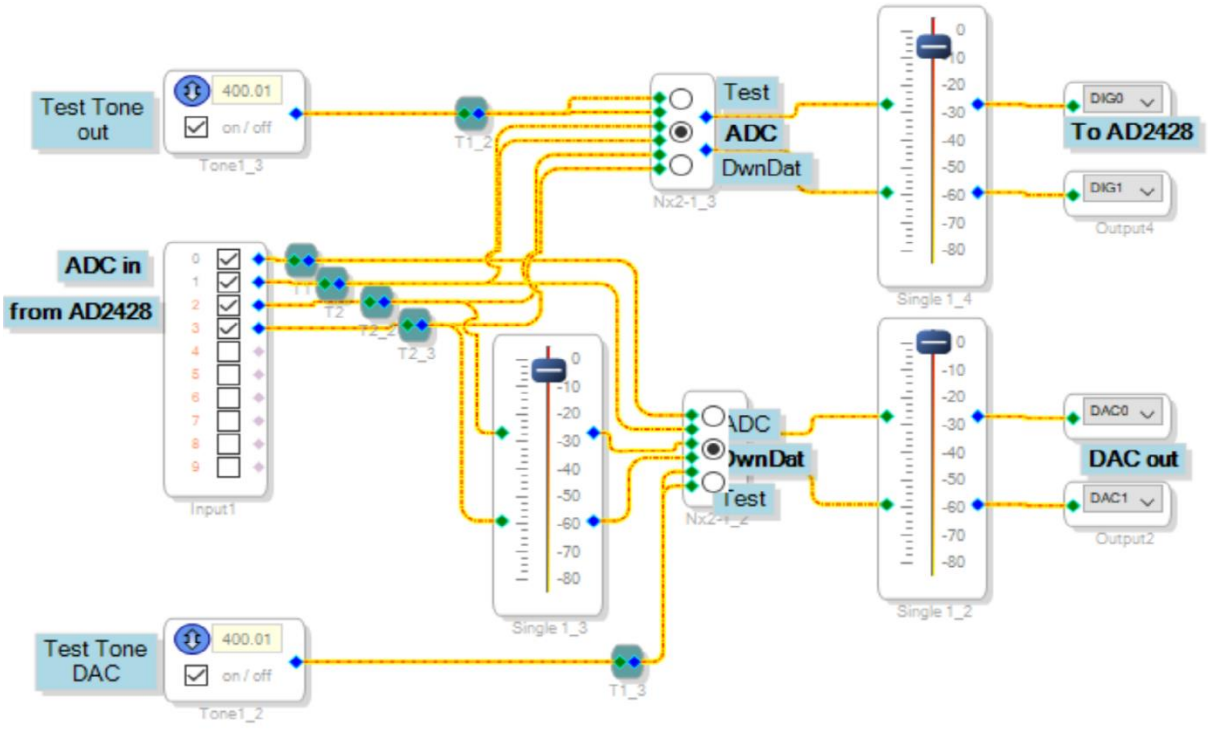


Figure 29 Master node ADAU1761 example (`CSP_EVM_a2b_master_ADAU1761.dspproj`)

The ADAU1761 is set up with the correct divider for the 12.288 MHz MCLK, which is 256 times Fs (Fs = 48 kHz).

On the master node the AD2428 uses the BCLK as an input, along with the FSYNC (or FLCK). For correct AD2428 the frame sync is the important clock, it multiples that clock up by 1024 to create the A²B network clock.

The bit clock on the master node is only used for clocking I²S data in and out of theAD2428.

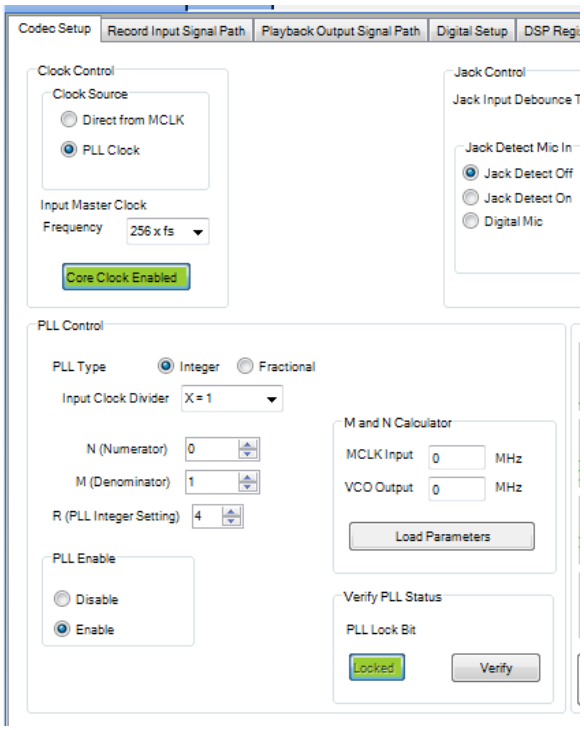


Figure 30 Master node ADAU1761 clock setup

For complete details on the ADAU1761 setup please see the datasheet and user guide. For the I²S interface setting it to Master mode is the key setting.

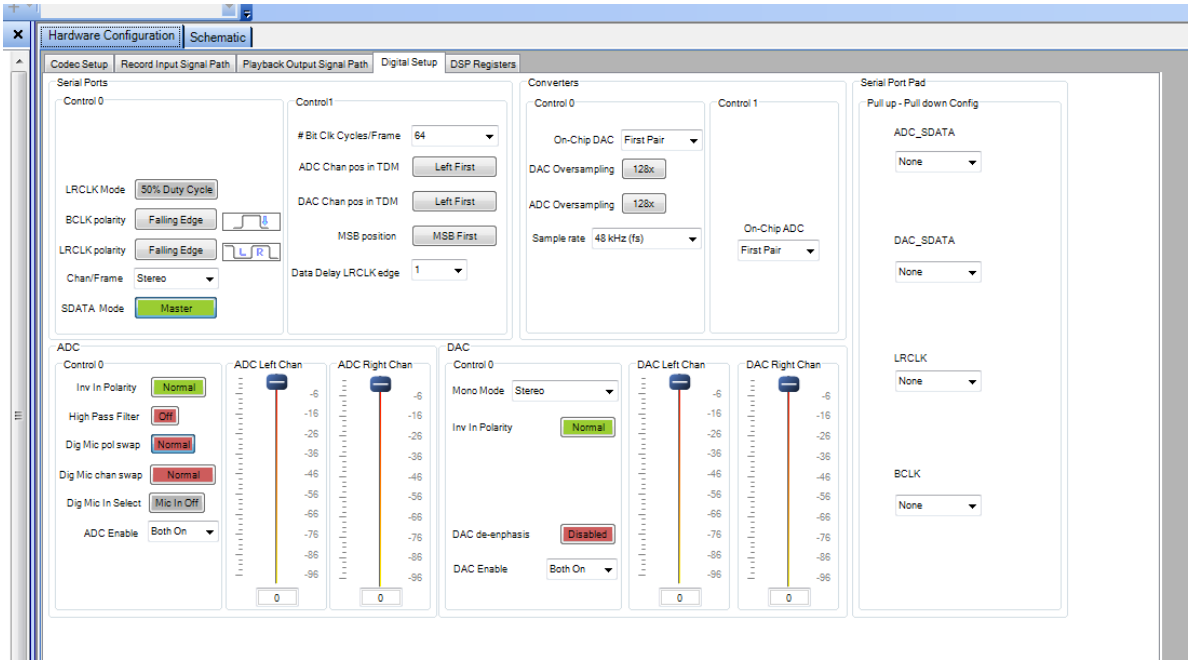


Figure 31 ADAU1761 serial port setting on EVM master node

Once the ADAU1761 has been set up the way you want you need to create the XML file that is used to program the ADAU1761 at boot up. Full details are in the SigmaStudio guide(s), but the summary is

- *Link Compile Download* (hardware doesn't need to be active)
- Copy the Capture window left side contents to the right side (hit the << button on the right if it's not visible)
- Save the right side window to an XML file
- Return to the A²B schematic and set the master node ADAU1761 to use the XML file

If you haven't already, use the Streams configuration to route audio data between the nodes in your system.

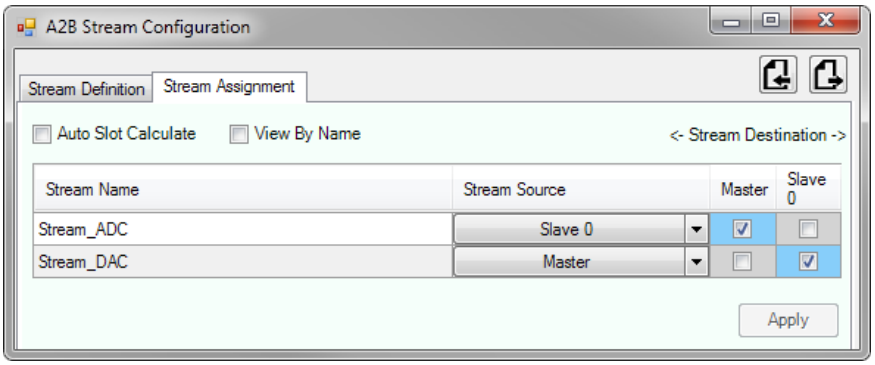


Figure 32 Simple stereo stream routing for two EVM example

At this point the A²B schematic is ready to run. If you're using the canned example you would have skipped all of the above. Hit the *Link Compile Download* and the two AD2428 nodes should turn green. If you get an error message then check that power is on to all of the boards and they are cabled correctly. If you made new diagrams then recheck your work.

7.1.1 THINGS THAT MIGHT GO WRONG

If there is a bug in your ADAU1761 schematic for the master node then it probably won't create the I²S clock(s), and the master node AD2428 will be unhappy. You may want to run your ADAU1761 schematic first and probe the I²S signal to make sure you set up I²S correctly.

Once the A²B schematic is running you can open the ADAU1761 schematic and dynamically change its controls and have them reflected in the running hardware. If opening the ADAU1761 schematic for the first time you should hit the *Link Compile Connect* button.

You can also work with a slave node's ADAU1761 in a similar manner to adjust its settings on the fly.

Very occasionally the USBi and/or SigmaStudio may get a little lost and not make your changes, or sometimes report it can't find the USBi. A power cycle of the A2B master and unplugging the USBi

from the host PC usually clears that up, sometime restarting SigmaStudio is needed too. This does not happen very often (we see it maybe once in a 100 runs where we're changing things).

It's also been noticed that if the A²B application is running and the host PC goes to sleep it can sometimes shut down the A²B target. Exit from SigmaStudio if you want to leave things going for extended periods.

8 SHARC AUDIO MODULE (SAM) AS A²B MASTER

The SC589 based SAM is a low cost development board from Analog Devices. At the time of writing of this manual revision there is not a lot of information available from ADI to simplify use of A²B with the SC589.

While the procedures described in ADI provided A²B software package documents theoretically explain the process of using SC589 with A²B, in real life it can be a bit, hmmm, let's all it obtuse. This situation will hopefully change when ADI releases the A²B based amplifier board.

The diagrams and software described in this section are available from Clockworks, if you don't find them on the support site please contact us.

Your first stop should be the ADI Wiki to gather up the latest things that they have made available.

https://wiki.analog.com/resources/tools-software/sharc-audio-module

8.1 IF YOU DON'T HAVE THE A²B SOFTWARE PACKAGE…

you won't be able to actually develop A²B based applications. You can however run the precooked A²B network configurations that ADI has made available for SAM, as well as the ones that Clockworks has created.

These precooked configurations can be used with your own custom SC589 software that you develop, you are restricted in that you can't change the topology, sample rates, routing, etc.

When the SAM Bare Metal package runs it simply loads a predefined sequence of I²C commands in to the master A²B node located on the SAM. This sets up all of the topology as well as programming other I²C devices like the commonly used ADAU1761 SigmaDSP or I²C devices likes the SSM3582 Class D amplifier chip used on the quad channel SAM FIN.

Once the A²B network is running the SAM software simply reads/writes data from/to the I²S port connected to the AD2425W A²B transceiver chip. For applications that don't need the sophisticated fault and error detection that A²B offers, this may be more than adequate.

If the hardware configuration might change – which after all is the whole point of the flexibility offered by A²B – you will need the tools[7].

8.2 USING THE ADI A²B TOOLS WITH SAM – HARDWARE AND SOFTWARE SETUP

The steps are as follows for a basic SAM plus Clockworks A²B EVM setup. Audio is just exchanged between the two boards in this example.

8.2.1 HARDWARE SETUP SUMMARY

Details of the hardware and boards can be found in their respective user manuals. Three USB ports are needed, keep that in mind if your laptop is USB challenged. The tools run on Windows 7 and later.

- Connect the ICE1000 to the SAM JTAG port and your PC
- Connect the USBi to the SAM USBi port (careful, it's just a header, so verify where pin 1 is) and your PC.
- Connect a UART to USB adapter to the UART header on SAM. Normal baud rate is 115,200.
- Connect audio sources to the SAM Line in jack and the A²B EVM line in jacks.
- Connect amplified speakers to the SAM line out jack.
- Connect headphones or amplified speakers to the A²B EVM headphone out jack.

[7] Or hire developers that use the tools to provide that piece of your product's software.

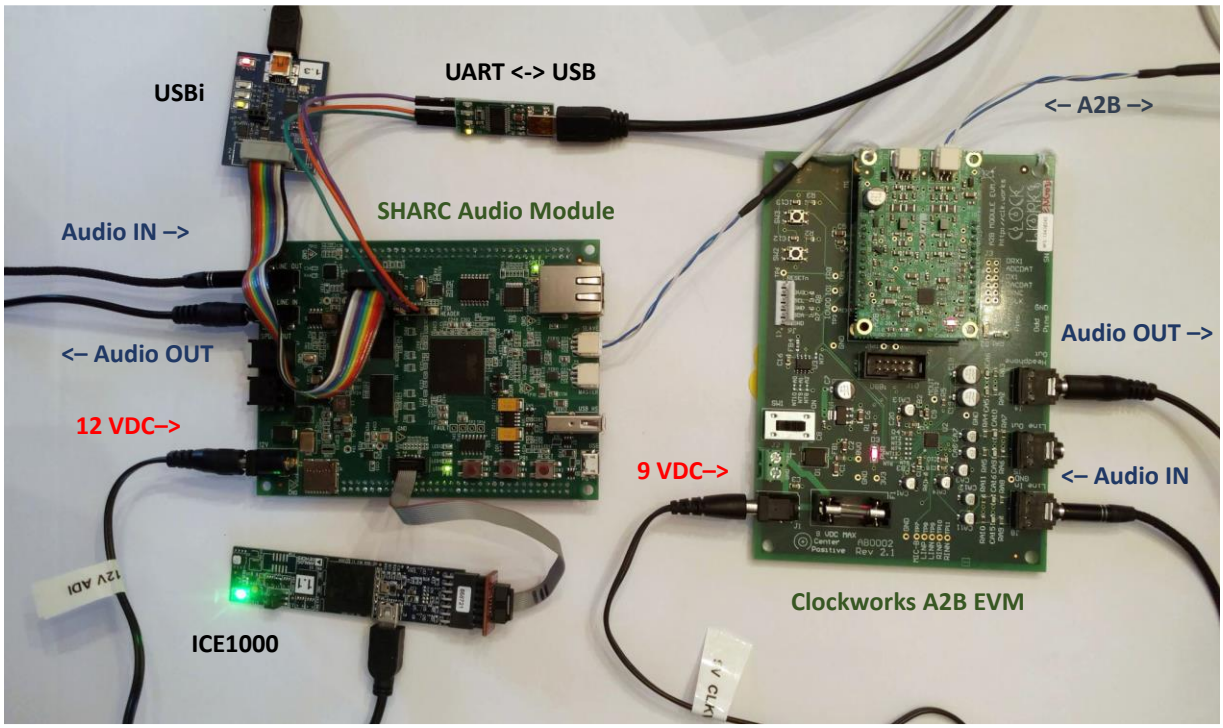


Figure 33 SAM and A2B EVM hardware setup

Do not connect 12V to the Clockworks A²B EVM board as the power is connected directly to the AD2428W on the A²B module and if you put more than 9V on the AD2428W chip's supply pin all of the magic smoke will be released.

8.2.2 SOFTWARE SETUP SUMMARY

There are a number of tools that are used in the process of creating A²B based applications. This manual is not the place to learn about them.[8]

- CCES. When used with ICE1000 and SAM the free version can be used.
- SigmaStudio
- A2B add on for SigmaStudio (including API)
- Terminal emulator for the UART connection
- SAM Bare Metal Framework
- A pre-cooked SC589 bypass program or one created from the wizard (requires editing)

[8] Are we lazy? Err, maybe. Seriously though the tools are always getting updated and if the details were copied in to here we would be forever updating this document, which would seriously cut in to the time available for sampling more IPAs.

8.3 BASIC SAM EXAMPLE WITH A2B EVM SETUP IN SIGMA STUDIO

This example exchanges audio between the two boards shown in the setup of Figure 33.

8.3.1 SIGMASTUDIO SCHEMATIC

Open the file `SAM-EVM.dspproj` in SigmaStudio[9] or create the equivalent on a new blank sheet.[10]

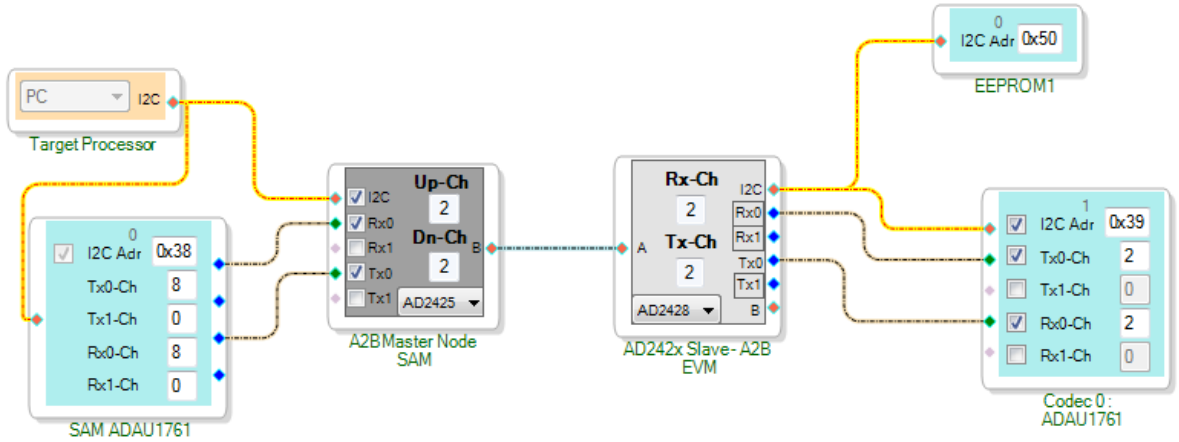


Figure 34 SAM with A2B EVM schematic

If starting from a blank schematic add the USBi connection on the Hardware Configuration tab. For SAM it looks like Figure 35.

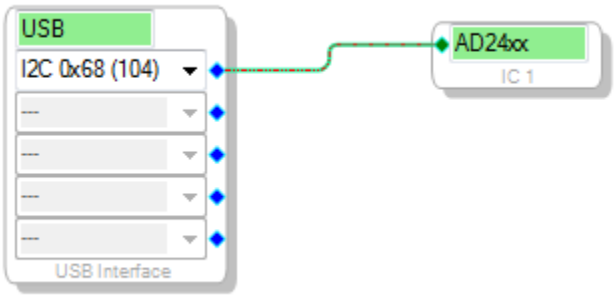
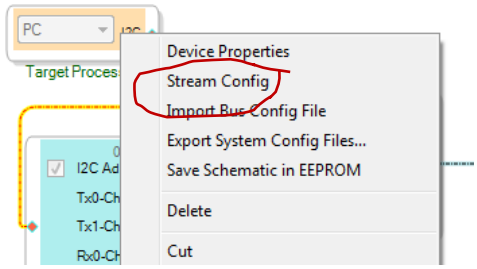


Figure 35 Sigma Studio Hardware Configuration tab for use with SAM and USBi

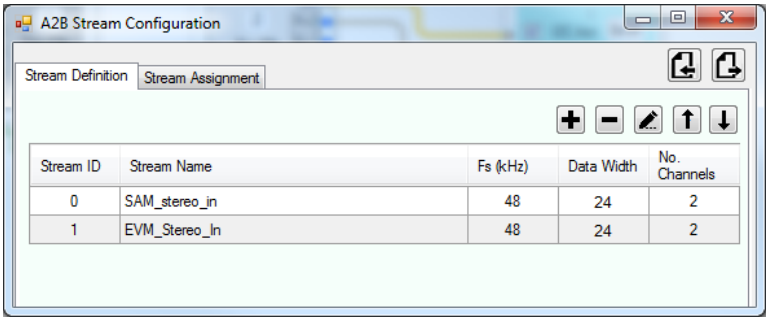
[9] A²B extensions must have been previously installed.
[10] If starting from scratch please see the information about A²B chip setup in Section 6.1.1.

The Stream capability of the later chips makes it easier to set up and manage audio data than using the manual configuration of the first generation chips. These next steps explain the process; the provided example already has this set up this way.

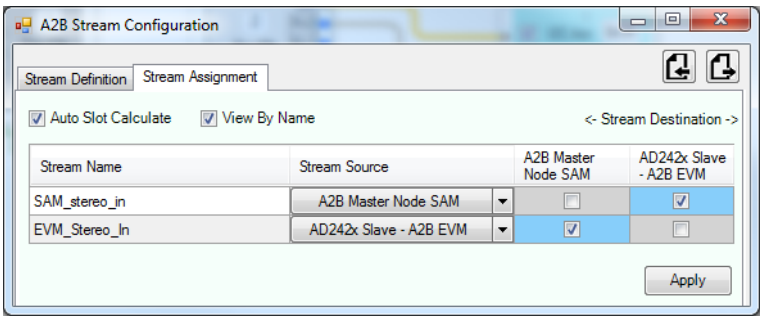
Right click on the Target processor block and select Stream Config:



Select the **+** button on the A2B Stream Configuration dialog and add two streams with 2 channels each:



Then select the stream assignment tab:

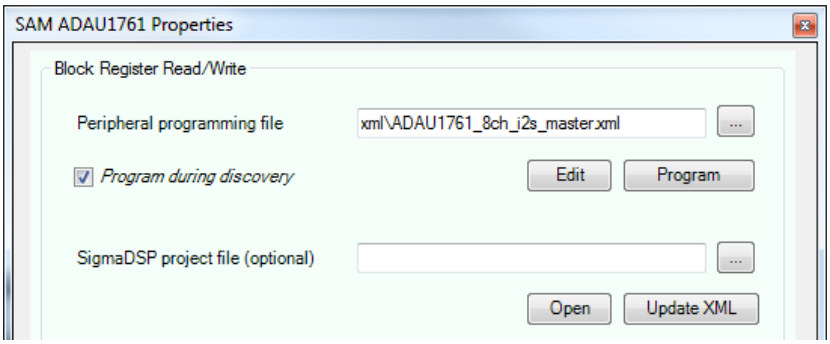


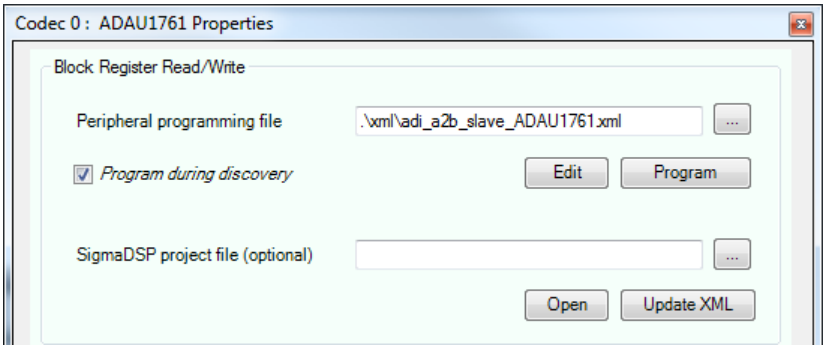
Make sure Auto Slot Calculate is selected, and View by Name[11] is usually more informative then viewing by ID. In this example we're swapping audio between the two nodes.

Next set up the programs that get run on the two ADAu1761 in the system (one on SAM, one on EVM). They run different programs. The process of creating the `.xml` files from SigmaStudio is covered in

[11] If the names of the AD24xx nodes are changed it's been noticed that this dialog can be a little slow to refresh the names. Clicking on a Stream Source seems to sync it back up.

the SigmaStudio documentation as well as in the A²B specific documents in the A²B add-on for SigmaStudio. The latter document also explains running and debugging ADAU1761 programs from within this setup.





If building an example from scratch check the I2S related settings for the EVM as described in Section 6.1.1.

Before doing the Compile Download & Run you must first load a pass through program on the SC589.

8.3.2 SAM SC589 BYPASS PROGRAM

There are some ADI provided examples that can be used for this step, or the SAM Wizard can be used to generate a basic pass through program.

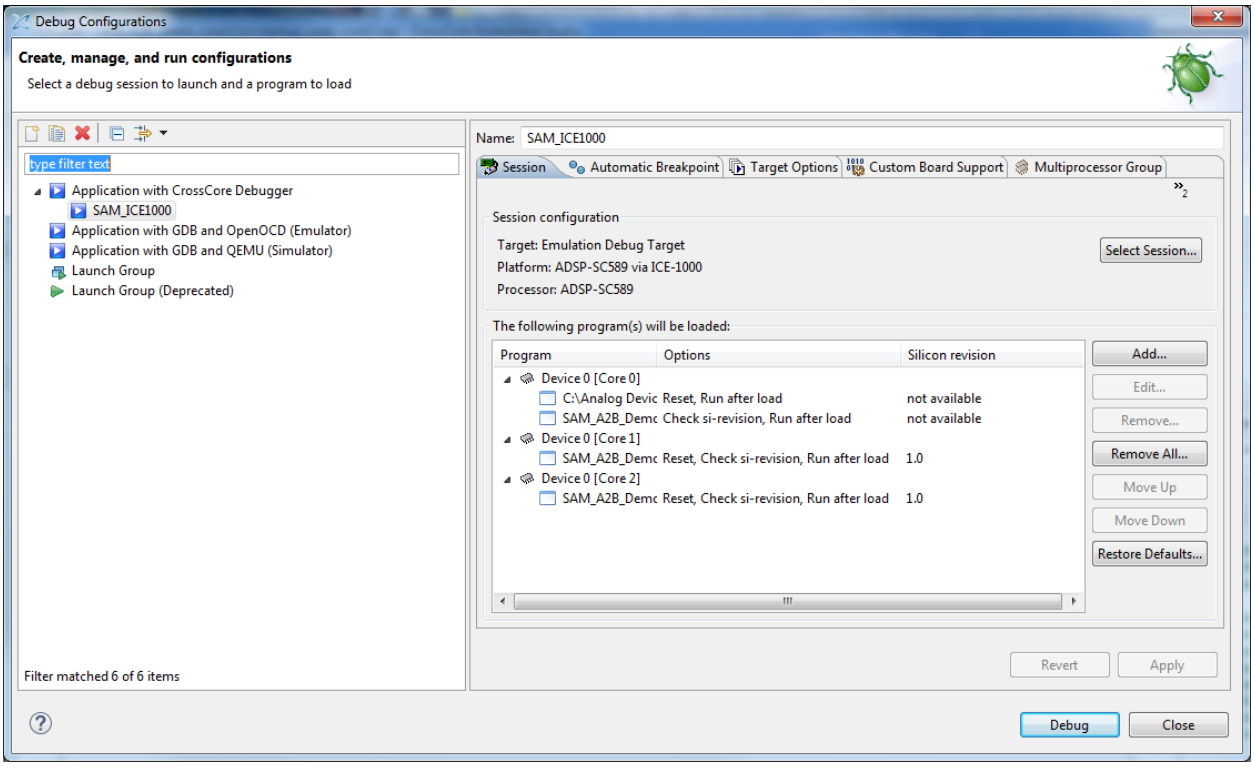
To edit the bare metal framework program, open up `common/audio_system_config.h` and make these changes[12]:

```
/*****************************************************************************
 * 3. Select an audio processing framework to use (only select one)
 *****************************************************************************/

// Standard audio processing framework (SHARC Audio Module
// and SHARC Audio Module + Audio Project Fin)
#define AUDIO_FRAMEWORK_8CH_SAM_AND_AUDIOPROJ_FIN          FALSE

// Audio processing framework for use with the automotive daughter board
#define AUDIO_FRAMEWORK_16CH_SAM_AND_AUTOMOTIVE_FIN        FALSE

// Bypasses the ADSP-SC589 so I2S signals routed directly between ADAU1761 and
```

[12] Thanks go to Dan Ledger at https://www.pathcollaborative.com/ for these code snippets.

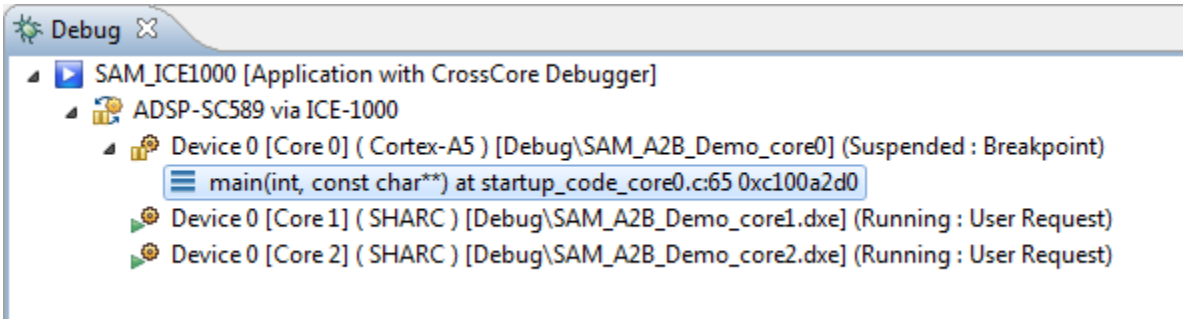
```
// A2B controller (GPIO4 = LED on this board)
#define AUDIO_FRAMEWORK_A2B_BYPASS_SC589                    TRUE    <--- If you're using SS, set
this to TRUE to bypass the SHARC

/*****************************************************************************
 * 4. Select whether or not to enable A2B in the framework
 *****************************************************************************/

#define ENABLE_A2B                                          TRUE

#if (ENABLE_A2B)
/**
 * If A2B is enabled, select the role that this SHARC Audio Module board
 * will play (TRUE = master node, FALSE = slave node)
 */
    #define A2B_ROLE_MASTER                                 TRUE    <--- If this SAM board is a
slave node on A2B bus, set to FALSE
```

If importing an existing project in to a CCES workspace verify that the Debug configuration is set up correctly[13].

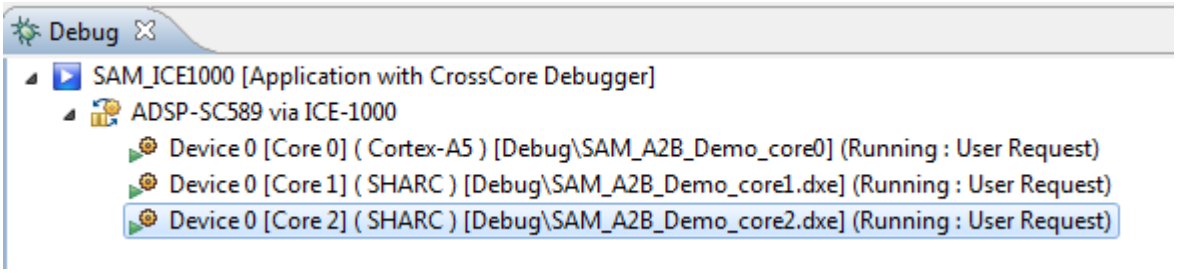


After you select Debug the code will be compiled (if needed) and if running with the stock CCES behaviors the Debug perspective will open, followed by the code loading on the three SC589 cores and then each stopping at the entry breakpoint:

[13] We skipped a lot of steps here, see the tutorials on SAM and CCES if this looked like magic to you. The good news is CCES is more or less standard Eclipse. The bad news is CCES is more or less standard Eclipse.



For each core select it and hit the run button, after doing that will all 3 it will look like:



And depending on the code being run the ADI framework startup messages will be displayed:

```
00:00:00.000 [INFO - ARM]  Baremetal Framework (version 2.0.0) for the ADI SHARC Audio Module

00:00:00.000 [INFO - ARM]  System Configuration:

00:00:00.000 [INFO - ARM]    Processor cores running at 450.00 MHz

00:00:00.000 [INFO - ARM]    Audio sample rate set to 48.00 KHz

00:00:00.000 [INFO - ARM]    Audio block size (per channel) set to 32 samples / frame

00:00:00.000 [INFO - ARM]  Configuring the SRU - ADAU1761 is the I2S master

00:00:00.000 [INFO - ARM]  Configuring the ADAU1761
```

The SC589 and the USBi both want to be I²C bus masters. Therefor the pass through program should not attempt I²C transactions as it will lead to something going south.

The system is now ready to have A²B operation be controlled from Sigma Studio over the USBi connection.

8.3.3 CONTROLLING FROM SIGMA STUDIO

Turn down the output level on your audio output device as a mistake in setting can result in full scale output noise. Set the source output level to around 50%, assuming some sane line level source is being used.

Select the Compile Link Download button in SigmaStudio. All of the AD242x node blocks should turn green. If you get an error message recheck your USBi setup, A²B wiring, and that the boards have power.

SigmaStudio should indicate its communications with the SAM board with a few messages/second. If that's happening but you get no audio confirm that the CCES session is still active[14] as if the SC589 isn't running then audio data will not be moved around.

When in doubt power cycle everything and start over. Very occasionally the USBi will go off in to the weeds and need to be unplugged from the PC and plugged back in. When in doubt check Device Manager for the USBi device (in the USB section).[15]

Bring up your audio output device volume. With two audio sources and two audio monitoring devices you should hear both sources. If one or both is highly distorted review the I²S settings as discussed in Section 6.1.1. If you hear clipping on loud parts your source is most likely overloading the inputs, turn down the audio source output level.

At this point you've now verified that your A²B topology works.

8.4 CREATE A PROGRAM THAT SETS UP THE A2B OPERATION THE SAME AS SIGMASTUDIO SCHEMATIC.

At some point in the far future, but before our Sun becomes a red giant, the SAM Project Wizard will have an option to generate a Bare Metal Framework project that allows direct selection of the Clockworks A²B EVM board as part of the A²B network. Until then though you'll need to slog through the details of all of this.

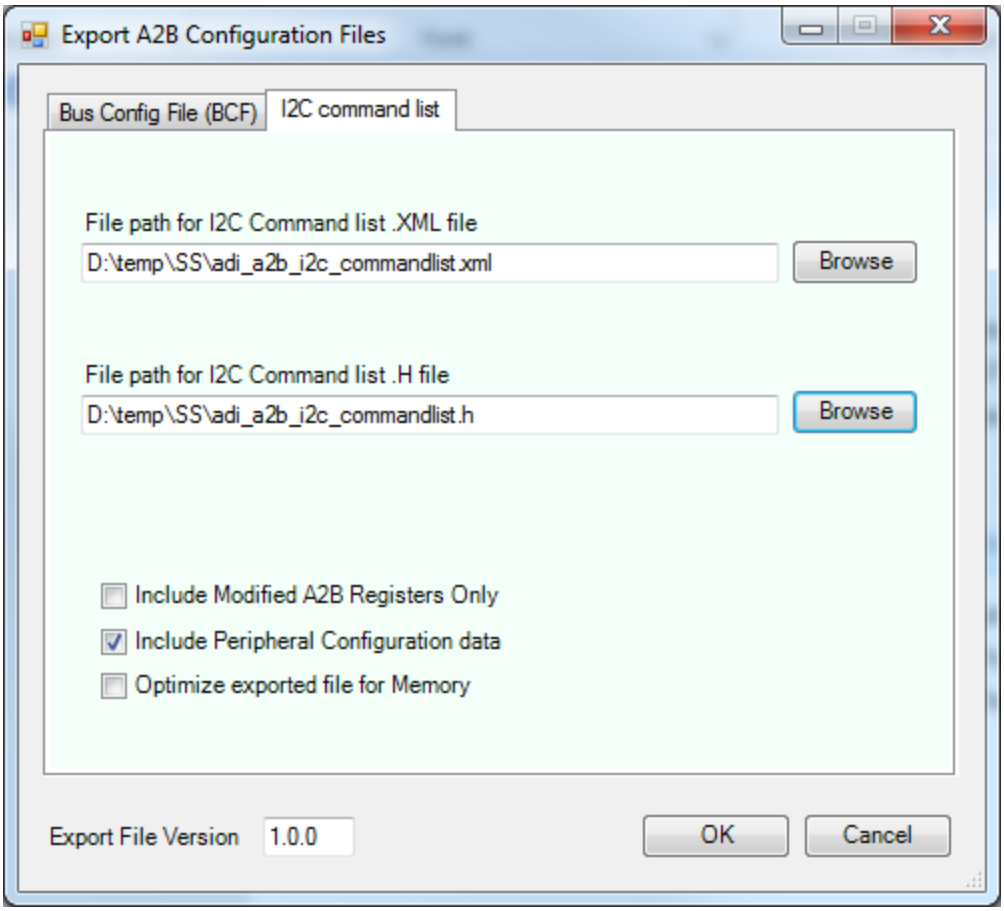
8.4.1 EXPORT FROM SIGMASTUDIO

Right click on the TargetProcessor icon and select the export option and the I²C tab. Fill in the target file locations; this can be pretty much anywhere as you'll copy them in to a CCES created Bare Metal Project Wizard created project. Select the Include Peripheral Data checkbox.

The dialog is a little deceiving, it needs a complete path and filename, not just the path (you'll get an access denied or similar message if you just provide a path). Best way to avoid issues is to use the browse button.

[14] Actually there's no real easy way to know this as you can quit CCES and the SC589 would still keep running. If the loaded program is printing to the console or flashing LEDS then that would be a clue its running. When in doubt restart. We should note we've never seen the SAM quit – it's more like a "oops I unplugged it for a second without realizing it" type thing.

[15] The appearance in device manager is the same if you use the USBi clone available from SushiBits on Tindie. Both ADI's USBi and the clone via Tindie were used when this manual was developed.



8.4.2 AND NEXT…

We're still evaluating what makes the most sense to take the exported file and glom it in to one of the Project Wizard created projects.

The SAM to SAM template is the closest starting point for a stereo audio application.