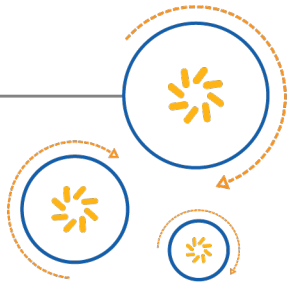
Qualcomm Technologies International, Ltd.



ADK 4.3 I²S

User Guide

80-CF421-1 Rev. AA

November 6, 2017

Qualcomm Technologies International, Ltd. (formerly known as Cambridge Silicon Radio Limited) is a company registered in England and Wales with a registered office at: Churchill House, Cambridge Business Park, Cowley Road, Cambridge, CB4 0WZ, United Kingdom.
Registered Number: 3665875 | VAT number: GB787433096

Revision history

Revision	Date	Description
AA	November 2017	Initial release. Alternative document number CS-00406801-UG Branched from CS-00301278-UG for ADK 4.3.

Contents

Tables

Figures

1 I²S operation in ADK applications - overview

I²S operation can be configured to suit various I²S audio devices and allows initialization as well as volume control through the I²C interface. Configuration is set using Sink Configuration Tool.

This document provides an example configuration for use with Analog Devices SSM2518 external amplifier device.

NOTE Settings for the SSM2518 amplifier are used to explicitly show how to configure the sink application from scratch. These settings are also hard-coded in the Sink application and can be enabled by setting **Plugin Type** to **Qualcomm SSM_2518 I2S Development** > **Board**, see How to configuration I²S operation.

2 How to configuration I²S operation

This section gives an annotated example for the configuration corresponding to the SSM2518 device for illustration purposes. However, if you wish to use this configuration, you can set **Plugin Type** to **Qualcomm SSM_2518 I2S Development Board** in the Sink Configuration Tool, see Figure 2-1.

NOTE Setting **Plugin Type** to **Qualcomm SSM_2518 I2S Development Board** implements the configuration settings described in:

- User-defined I²S configuration
- I²S settings
- I²C control data settings

It is then not necessary to set any of the other options described in this section.

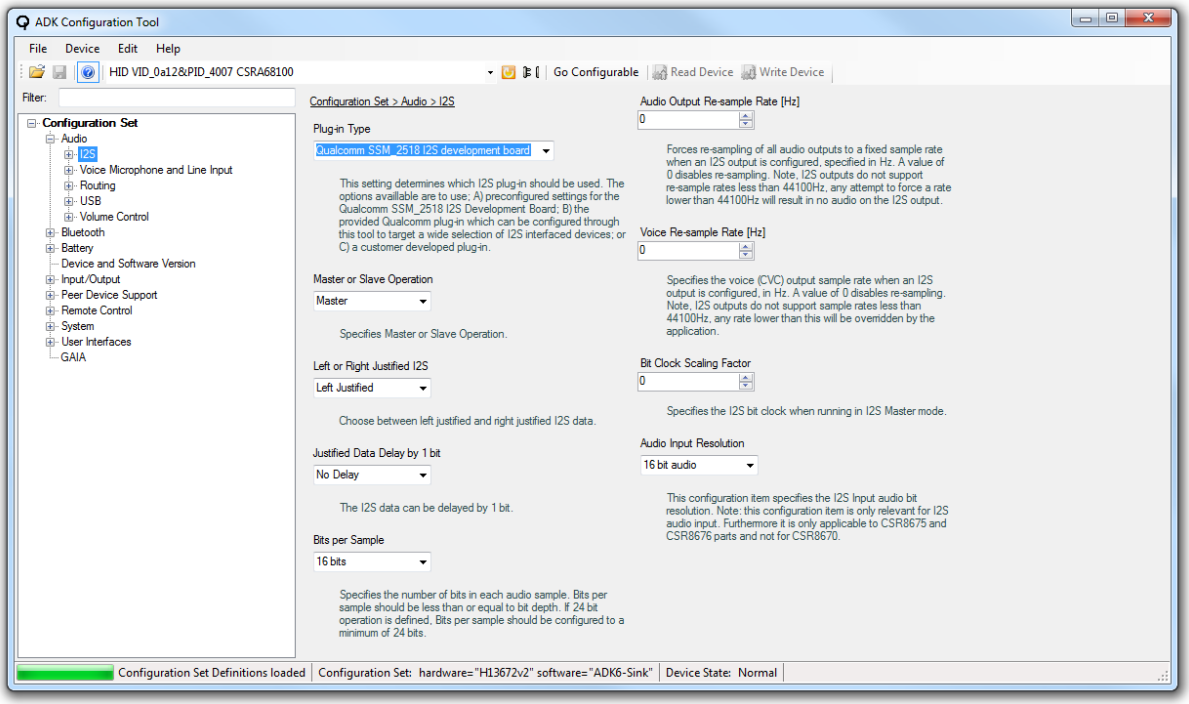


Figure 2-1 Analog devices SSM2518 plugin type

2.1 User-defined I²S configuration

To manually configure an external I²S amplifier, set the **Plugin Type** to **Qualcomm User-defined I2C control plugin**, see Figure 2-2.

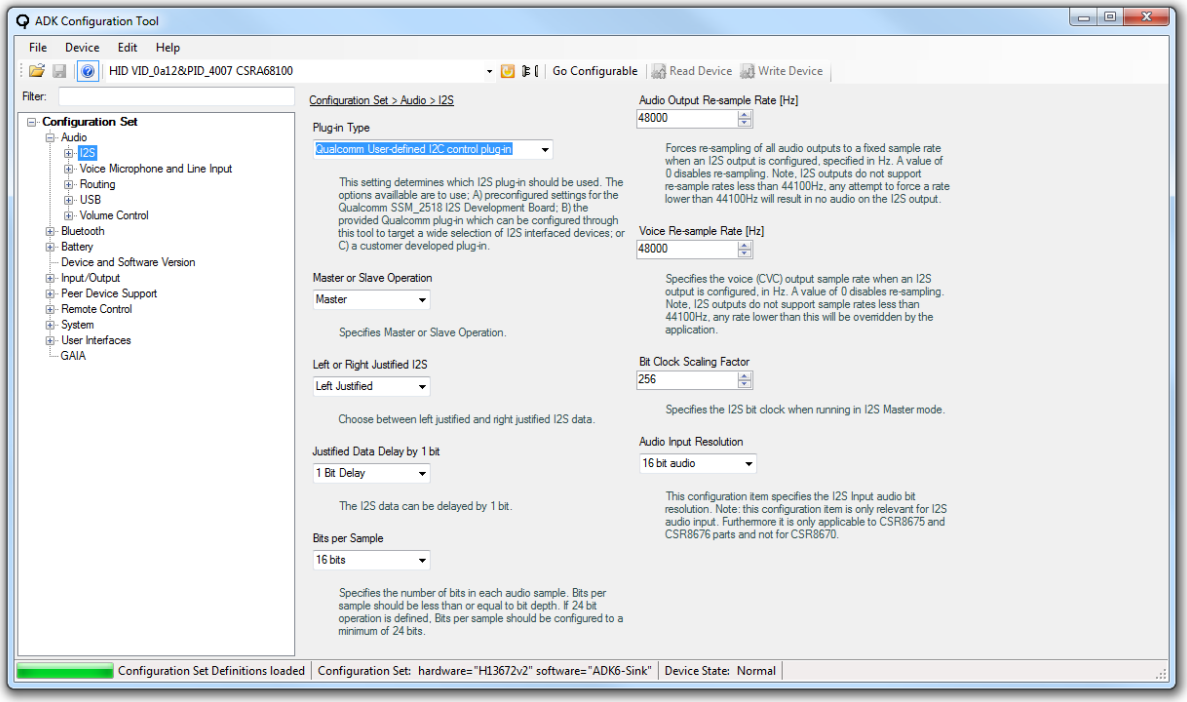


Figure 2-2 User-defined plugin type and associated I²S configuration items

2.2 I²S settings

Table 2-1 describes the various I²S configuration items that are available.

Table 2-1 Configurable data fields

Field	Description	SSM2518 Value
Plugin Type	Specifies which plugin to use, currently available options are: ■ User-defined using these configuration options ■ SSM_2518 I²S development board ■ Customer developed plugin	User-defined
Master or Slave operation	Whether the Sink application should act as a master or slave on the I²C bus. **NOTE** This implies the opposite setting for the external I²S amplifier.	Master
Left or Right Justified I²S	Specifies the format of the I²S audio data	Left Justified
Justified Data Delay by 1 bit	Specifies the format of the I²S audio data	1 Bit Delay

Table 2-1 Configurable data fields (cont.)

Field	Description	SSM2518 Value
Bits Per Sample	Specifies the number of bits in each sample of the I²S audio output data.	16 bits
Audio Output Re-sample Rate [Hz]	Specifies whether the music (A2DP, codecs, USB, wired audio) output needs to be resampled to a fixed rate. A value of 0 indicates that no resampling is performed. Otherwise this value specifies the required output frequency in Hz, for example 48000	48000
Voice Re-sample Rate [Hz]	Specifies whether the voice (cVc, tones, voice prompts) output needs to be resampled to a fixed rate. A value of 0 indicates that no resampling is performed. Otherwise this value specifies the required output frequency in Hz, for example 48000	48000
Bit Clock Scaling Factor	Specifies the scaling factor for the I²S bit clock. This overrides the bit clock frequency. For example: 256 x 48000 = 12.28 MHz If it is set to zero, the bit clock is calculated as follows for 16 bit 48 kHz audio: 16 x 2 (stereo) x 48000 = 1.536 MHz	256
Audio Input Resolution	Specifies the number of bits in each sample of the I²S audio input data.	16

2.3 I²C control data settings

To use many external I²S amplifiers, some commands must be sent over an I²C interface to power up and configure the device. Similar commands are also sent to shut down the device and change other settings, such as the volume.

2.3.1 External I²S device enable PIO

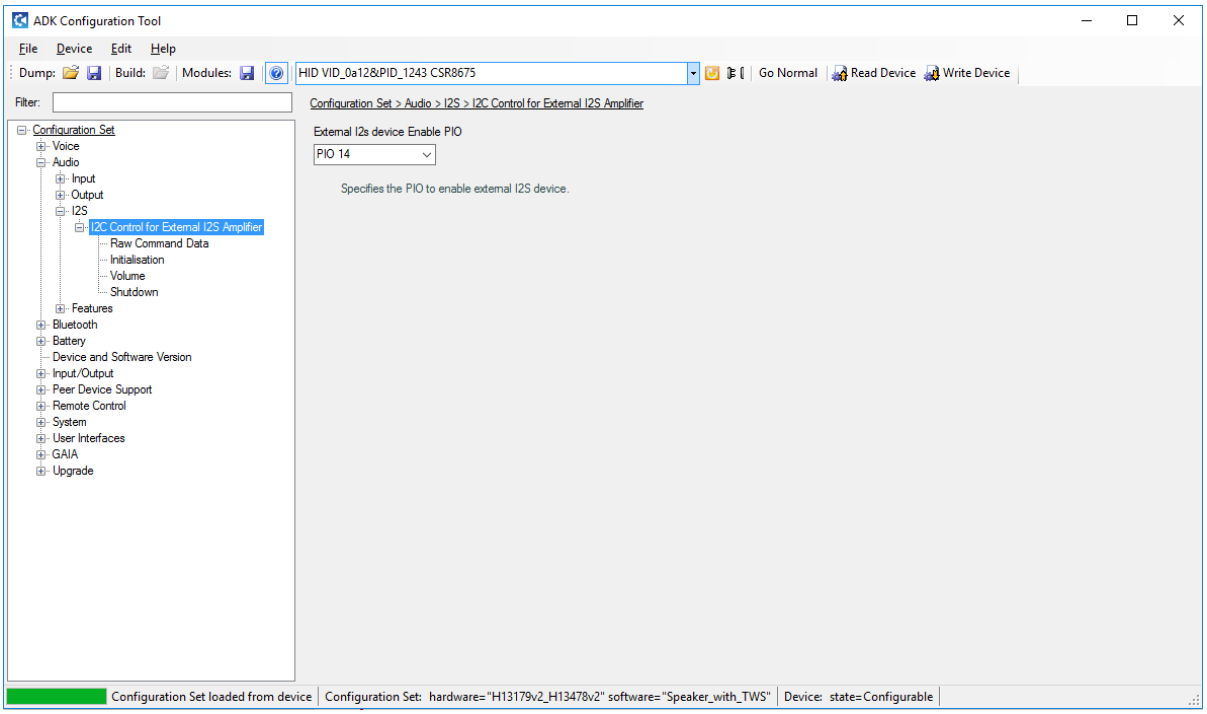


Figure 2-3 External device enable PIO

Figure 2-3 shows the settings page where an external I²S device can be enabled by selecting the appropriate PIO for the external device. When an H13117 reference external amplifier board is used, the External I²S device enable PIO must be configured to PIO14 to enable the external amplifier when the sink device wants to use it.

2.3.2 I²C raw command data

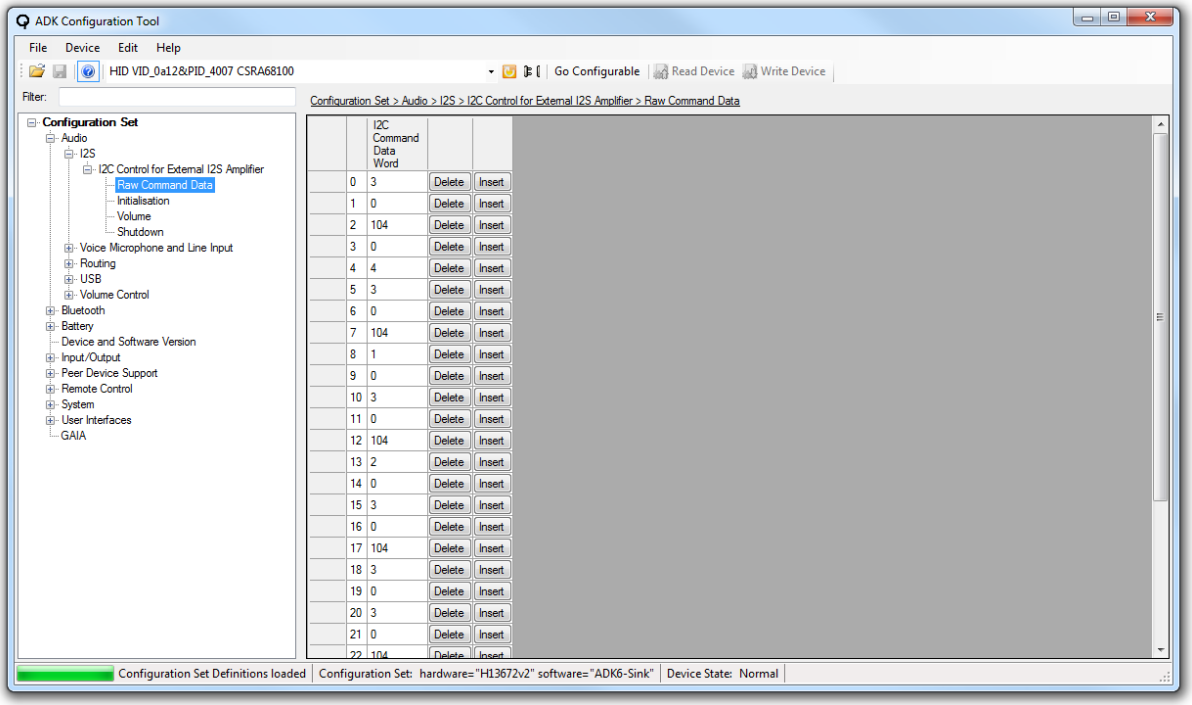


Figure 2-4 Raw command data

Figure 2-4 shows the settings page where the raw I²C commands are entered. Table 2-2 lists some example values. The commands are represented by groups of normally 5 octets, although they are entered as one long continuous stream as they can be longer than 5 octets. A worked example showing how to arrive at these values is given in I²C command data format.

NOTE The values entered into the configuration tool are decimal, despite being calculated and displayed in this document as hexadecimal.

Table 2-2 Example raw command data for the SSM2518

Packet Length	Volume Offset	I²C ID	I2CData[0]	I2CData[1]
0x03	0x00	0x68	0x00	0x04
0x03	0x00	0x68	0x01	0x00
0x03	0x00	0x68	0x02	0x00
0x03	0x00	0x68	0x03	0x00
0x03	0x00	0x68	0x05	0x40
0x03	0x00	0x68	0x06	0x40
0x03	0x00	0x68	0x07	0x00
0x03	0x00	0x68	0x09	0x98
0x03	0x00	0x68	0x07	0x01

Table 2-2 Example raw command data for the SSM2518 (cont.)

Packet Length	Volume Offset	I²C ID	I2CData[0]	I2CData[1]
0x03	0x01	0x68	0x05	0x40
0x03	0x01	0x68	0x06	0x40

2.3.3 I²C Initialization commands

Figure 2-5 shows how the number of I²C initialization commands is configured. The initialization commands are sent every time the external I²S amplifier is turned on. It is assumed that initialization commands begin from offset 0 in the raw command data. The example settings for the SSM2518 use 8 initialization commands, each 5 octets in length, for a total of 40 octets.

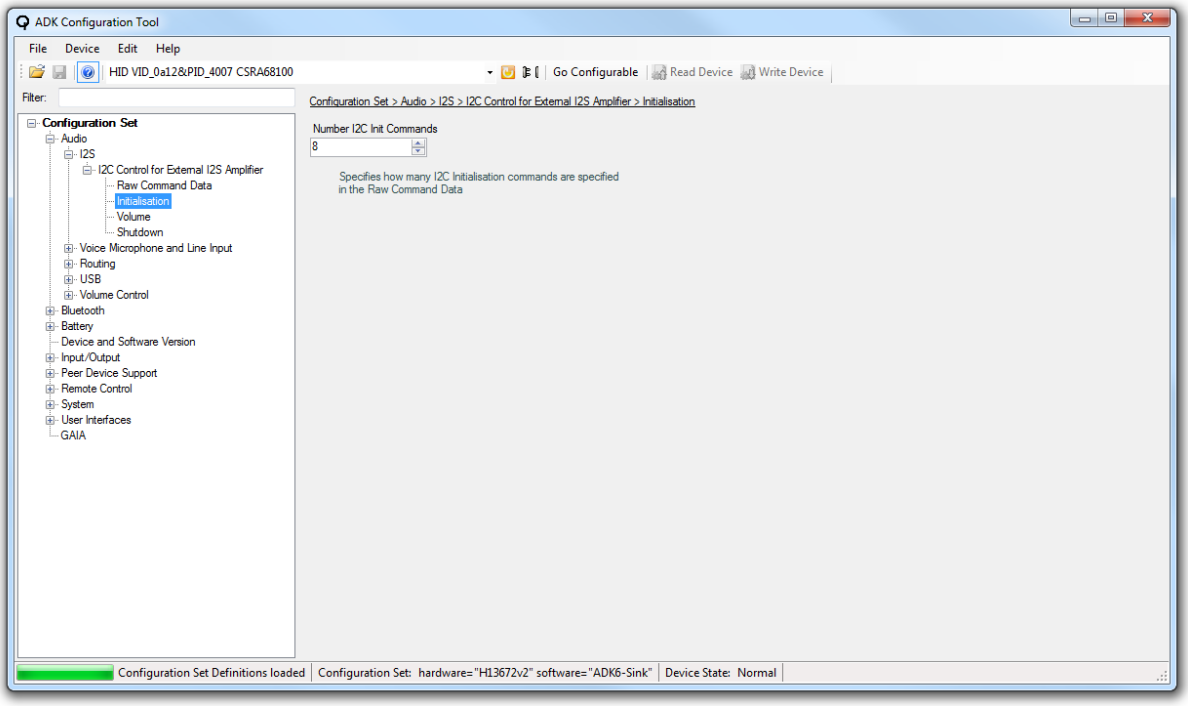


Figure 2-5 I²C initialization commands

2.3.4 I²C volume commands

Figure 2-6 shows how to configure the number of I²C Volume commands, and the offset in the raw command data at which they begin. Other parameters include the ability to set the number of bits used to represent the volume, and the minimum and maximum value to send. The volume is scaled automatically to this range. Table 2-3 describes the configurable fields and their example values for the SSM2518.

Table 2-3 Configurable data fields for I²C volume commands

Field	Description	SSM2518 Value
Number of I²C Volume Commands	Specifies the number of individual volume commands that are sent to the I²S capable device every time a volume change is made	2
Volume Commands Offset	Specifies the offset, in octets, of the first I²C volume command in the raw command data	45
Volume Number of Bits	Specifies the number of bits of the volume value in the I²C command	8
Volume Range Max	Specifies the maximum volume that will be sent to the I²S device to indicate maximum output level	0
Volume range min	Specifies the minimum volume level that will be sent to the I²S device to indicate minimum output level	255

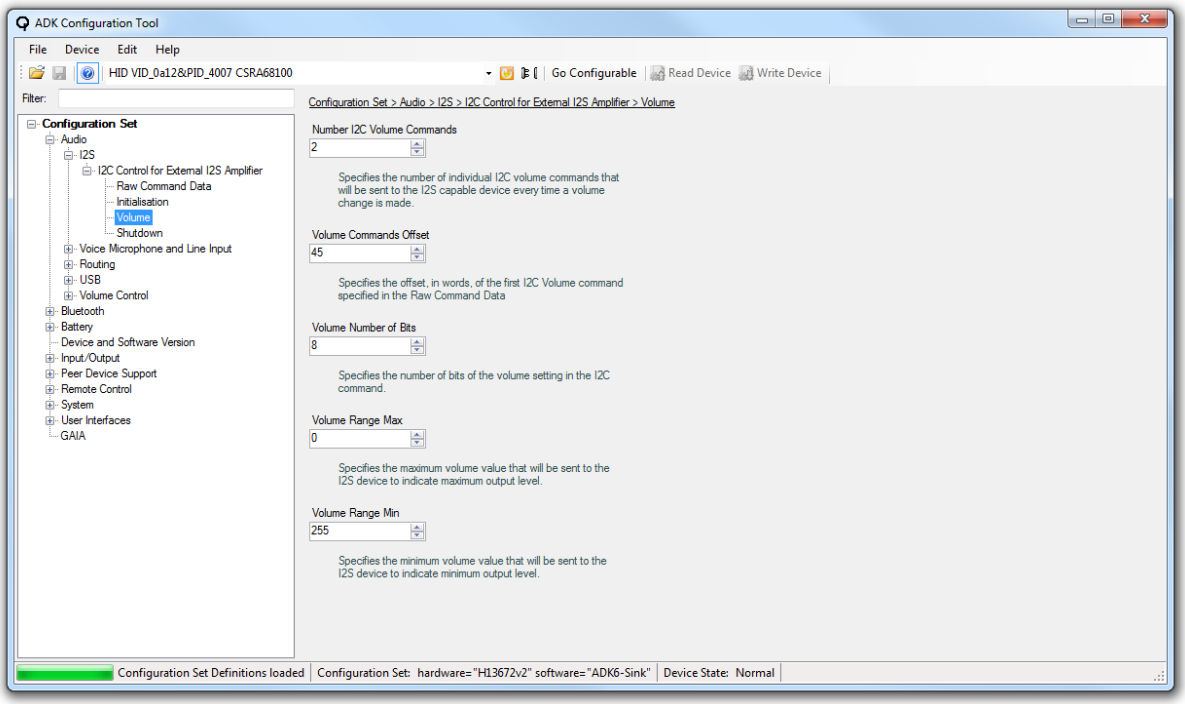


Figure 2-6 I²C volume commands

NOTE I²C volume commands are only supported in ADK 4.x. In ADK 6, all volume scaling is performed digitally by the DSP and there is no support for hardware volume control, and the I²C volume commands are never sent (even if configured).

2.3.5 I²C shutdown commands

Figure 2-7 shows how to configure the number of I²C shutdown commands, and the shutdown command offset in the raw command data. For the SSM2518, there is a single 5-octet shutdown command at an offset of 40 in the raw command data.

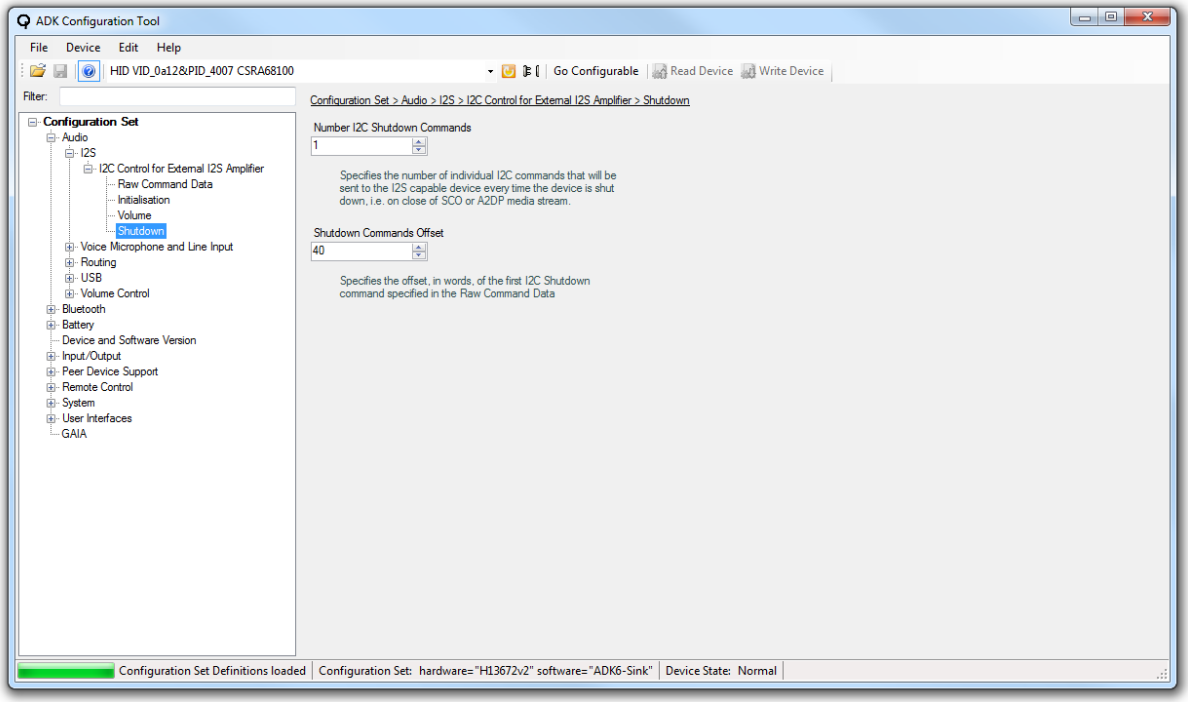


Figure 2-7 I²C shutdown commands

3 I²C command data format

The raw data structure of the I²C commands sent to initialize, configure, and shutdown an external I²S amplifier is described in:

- I²C packet length
- I²C volume offset data
- I²S packet data
- SSM2518 example

Table 3-1 shows some example I²C commands from the earlier SSM2518 data for reference.

Table 3-1 Example I²C commands

Packet Length	Volume Offset	I²C ID	I2CData[0]	I2CData[1]
0x03	0x00	0x68	0x00	0x04
0x03	0x00	0x68	0x01	0x00
0x03	0x00	0x68	0x02	0x00

3.1 I²C packet length

This is the length of the packet in octets. For example, an I²C initialization command packet of `0x68, 0x00, 0x04` has a packet length of 3.

NOTE `Packet Length` and `Volume Offset` are not included in the `Packet Length` calculation.

3.2 I²C volume offset data

This word is only applicable to data packets that set the volume/audio level output for the left/right channels. Its value is ignored for non-volume related packets.

Table 3-2 describes an example I²C volume packet.

Table 3-2 I²C set volume data

I²C ID	Set Volume Command	Volume Level
0x68	0x05	0x??

The volume level offset for the example volume packet as described in Table 3-2 would be 1 since the first data byte is always the I²C device ID. Data byte[0] is `0x05` (set volume command). Data byte[1] is the actual volume level.

The volume level is not necessarily 8 bits, its size is configured in the Volume section of the I²C control data settings, see I²C volume commands. If the volume level size is larger than 1 byte then the volume level offset indicates the start of the data to be replaced with the current sink or subwoofer volume level.

Table 3-3 Set volume command with 8 volume bits

Packet Length	Volume Offset	I²C ID	Set Volume Command	Volume Level
0x03	0x01	0x68	0x05	0x40

Table 3-4 Set volume command with 16 volume bits

Packet Length	Volume Offset	I²C ID	Set Volume Command	Volume Level[0]	Volume Level [1]
0x04	0x01	0x68	0x05	0xff	0xff

Table 3-5 Set volume command with 8 volume bits and 16 command bits

Packet Length	Volume Offset	I²C ID	Set Volume Command[0]	Set Volume Command[1]	Volume Level
0x04	0x02	0x69	0x01	0x06	0x40

3.3 I²S packet data

The packet data is an array of I²C command bytes whose length is specified by the `Packet Length` value. The first octet is always the I²C ID, that is, the address of the external chip on the I²C bus. This is usually given in the manufacturer's data sheet, though many chips have chip-select lines which allow some control over this address. The remaining octets comprise the data actually sent to the device, often a command-value pair.

Table 3-6 describes an example I²C initialization command.

Table 3-6 I²C set sample rate command

I²C ID	Set Sample Rate	Automatic Sample Rate Control
0x68	0x01	0x01

The complete packet and header information would comprise five words and be as described in Table 3-7.

Table 3-7 Set sample rate PS Key data

Packet Length	Volume Offset	I²C ID	Set Sample Rate	Automatic Sample Rate Control
0x03	0x00	0x68	0x01	0x01

3.4 SSM2518 example

The raw command data is split into tables of 5 octet chunks, as all the commands used in this example are 5 octets long. The commands are also split into separate tables for initialization, shutdown, and volume commands.

The number of initialization, shutdown, and volume commands are configured using the relevant settings page in the Sink Configuration Tool, see I²C control data settings, as are the volume and shutdown command offsets in the command data (initialization commands are assumed to start at offset 0).

Table 3-8 Initialization commands

Data	SSM2518 Register Name	Description
03 00 68 00 04	Reset_power_control	SPWN = 0 for normal operation MCS = 0010 256 x fs (Master clock select, value must match Word 10 in PSKEY_USR35) NO_BCLK = 0 for BCLK pin used as bit clock source RESERVED = 0 S_RST = 0 for normal operation
03 00 68 01 00	Edge_clock_control	ASR = 0 Automatic detection enabled EDGE = 0 No edge rate control RESERVED = 0
03 00 68 02 00	Serial_interface_sample_rate_control	FS = 0 (only required if ASR = 1 in register 0x01) SAI = 0 I2S left justified, or right justified stereo depending on SDATA_FMT) SDATA_FMT = 0 I²S standard, data is delayed by one BLCK cycle RESERVED = 0
03 00 68 03 00	Serial_interface_control	RESERVED = 0 BLCK_EDGE = 0 Rising BLCK edge used SLOT_WIDTH = 00 32 BLCK cycles per slot SAI_MSB = 0 MSB first LRCLK_POL = 0 Rising edge (normal) LRCLK_MODE = 0 50% duty cycle BCLK_GEN = 0 Bit clock from BCLK pin is used
03 00 68 05 40	Left_volume_control	Set left volume to 0 dB
03 00 68 06 40	Right_volume_control	Set right volume to 0 dB

Table 3-8 Initialization commands (cont.)

Data	SSM2518 Register Name	Description
03 00 68 07 00	Volume_mute_control	M_MUTE = 0 Normal operation L_MUTE = 0 Normal operation R_MUTE = 0 Normal operation VOL_LINK = 0 Normal operation DEEMP_EN = 0 De-emphasis disabled (normal operation) ANA_GAIN = 0 Matched to 3.6 V supply RESERVED = 0 AMUTE = Automute enabled
03 00 68 09 98	Power_fault_control	APWDN_EN = 0 Automatic power down disabled L_PWDN = 0 Normal operation R_PWDN = 0 Normal operation DAC_LPM = 1 Low power operation AMP_LPM = 1 Low power operation RESERVED = 0 AR_TIME = 10 40 ms autorecovery delay

Table 3-9 Shutdown commands

Data	SSM2518 Register Name	Description
03 00 68 07 01	Volume_mute_control	M_MUTE = 1 Master Mute L_MUTE = 0 Normal operation R_MUTE = 0 Normal operation VOL_LINK = 0 Normal operation DEEMP_EN = 0 De-emphasis disabled (normal operation) ANA_GAIN = 0 Matched to 3.6 V supply RESERVED = 0 AMUTE = Automute enabled

Table 3-10 Set volume commands

Data	SSM2518 Register Name	Description
03 01 68 05 40	Left_volume_control	Set left volume command (last byte ignored)
03 01 68 06 40	Right_volume_control	Set right volume command (last byte ignored)

4 I²S extra configuration - audio routing

I²S output is enabled using the Sink Configuration Tool by modifying **Configuration Set > Audio > Routing > Multi-channel Plugin** so that the **Digital Audio Interface Output 0** is enabled and configured to route the primary DSP output to I²S. When enabled, all audio output is redirected to an I²S capable device, including tones and voice prompts, see Figure 4-1 and Figure 4-2.

When CSR8670 and CSR8675 run Soundbar application, the Power down mode for the external audio amplifier stage can be managed using configurable PIOs depending on the hardware design.

For the Soundbar application to drive PIOs to switch off the external amplifier when going into Stand-by mode (that is, Limbo state) and switch it on while powering on, the configuration item **Amplifier Power Down by PIO** must be enabled. For detailed information about this configuration item, refer to the *ADK Sink Application User Guide*.

The PIOs used to generate the correct logic for amplifier Power and Mute signals are then configured as in I²S output PIOs.

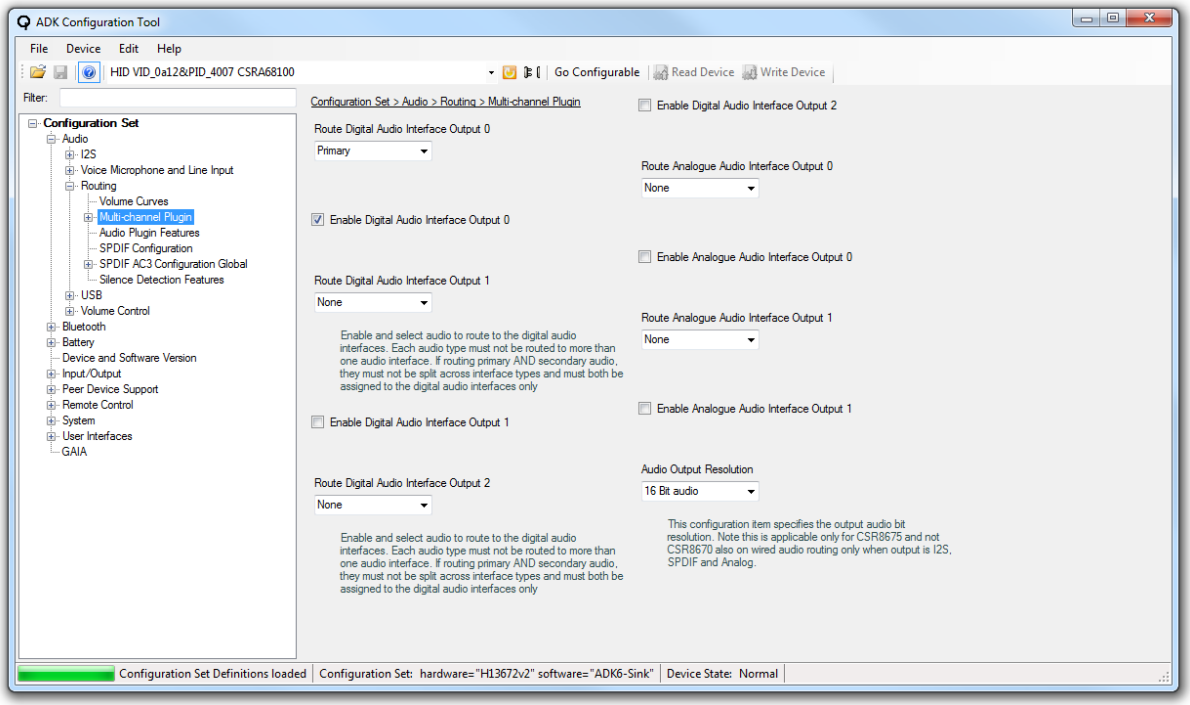


Figure 4-1 Configuring digital output 0

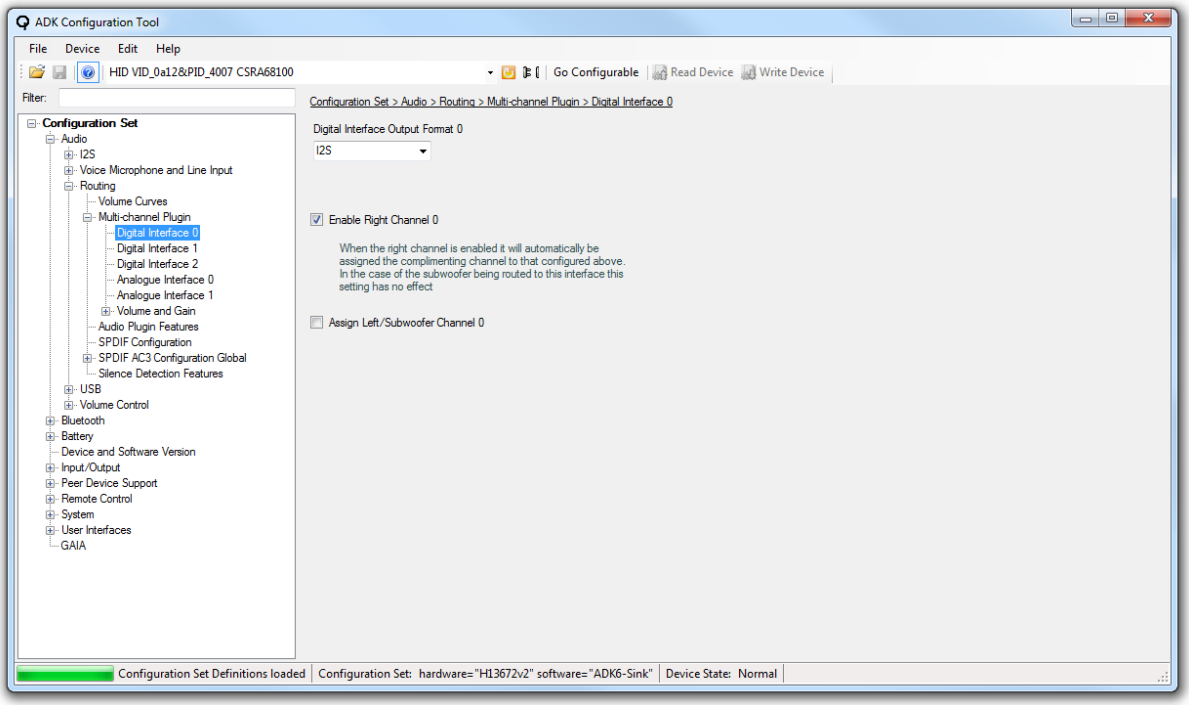


Figure 4-2 Configuring I²S audio on digital output 0

4.1 I²S volume control (ADK 4.x only)

Volume Scaling Method must be configured to **Hardware** for volume control commands to work using the I²C interface. This is because the raw I²S digital audio contains no volume data. Therefore, I²C commands must be sent to the I²S amplifier to configure the final output volume.

Volume Scaling Method may be configured to **DSP** to use digital gain rather than I²C commands to control the I²S volume. The **Hybrid** option is not supported for I²S outputs.

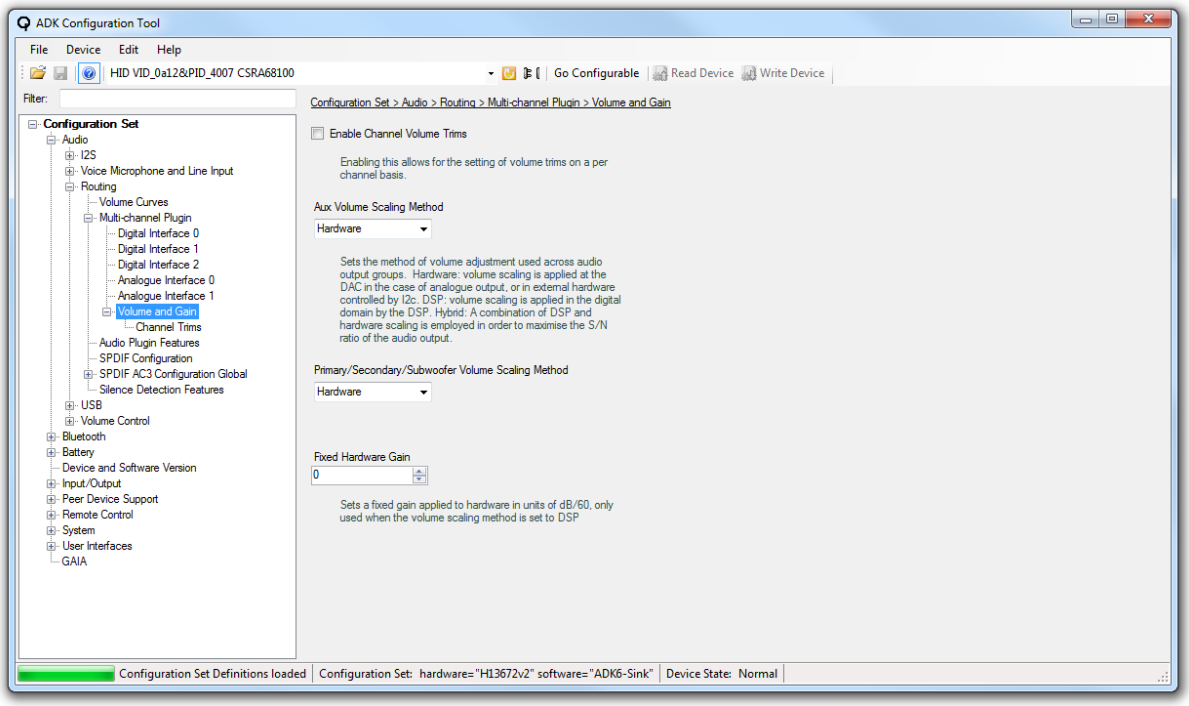


Figure 4-3 Volume control setting

NOTE I²C volume commands are only supported in ADK 4.x. In ADK 6, all volume scaling is performed digitally by the DSP and there is no support for hardware volume control and the I²C volume commands are never sent (even if configured).

4.2 I²S input PIOs (ADK 4.x only)

To use the I²S amplifier with CSR8670, the **SPDIF Input PIO** must be configured to **Disable**. With CSR8675 the **SPDIF Input PIO** may be enabled along with the I²S amplifier, provided the **SPDIF Output PIO** is configured as described in I²S output PIOs.

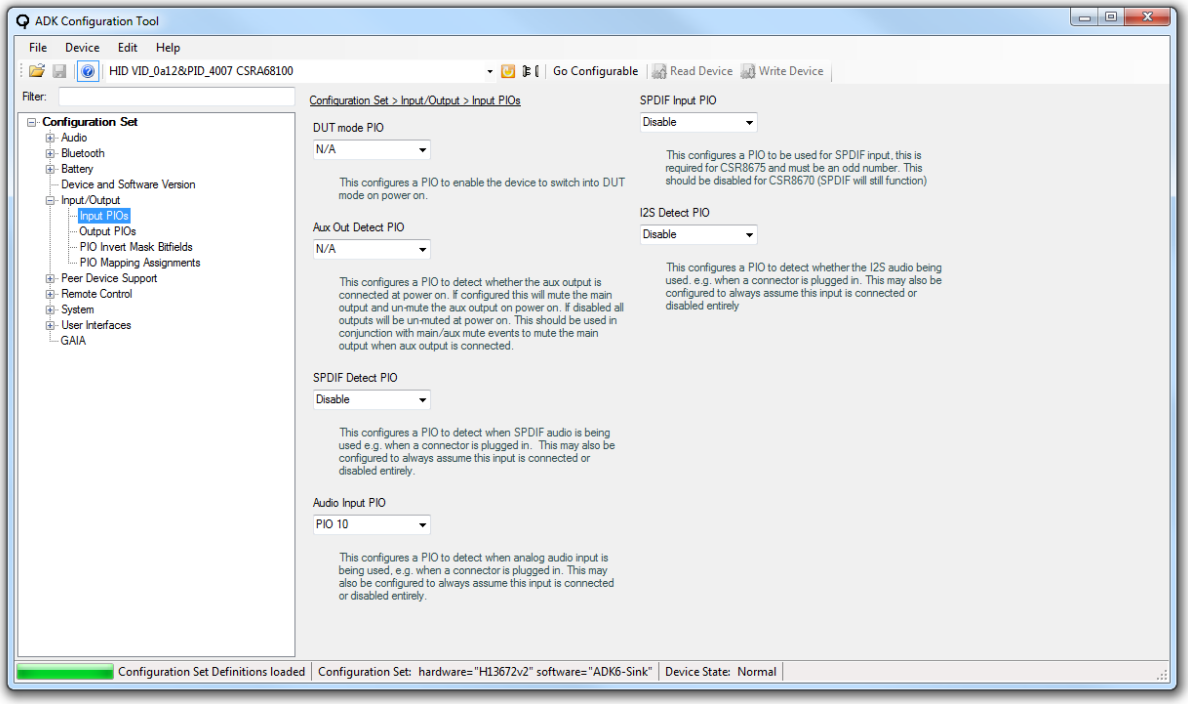


Figure 4-4 Input PIO configuration

4.2.1 I2S output PIOs

To use the I2S amplifier with CSR8675, the **SPDIF Output PIO** must be configured to **N/A**, see Figure 4-5.

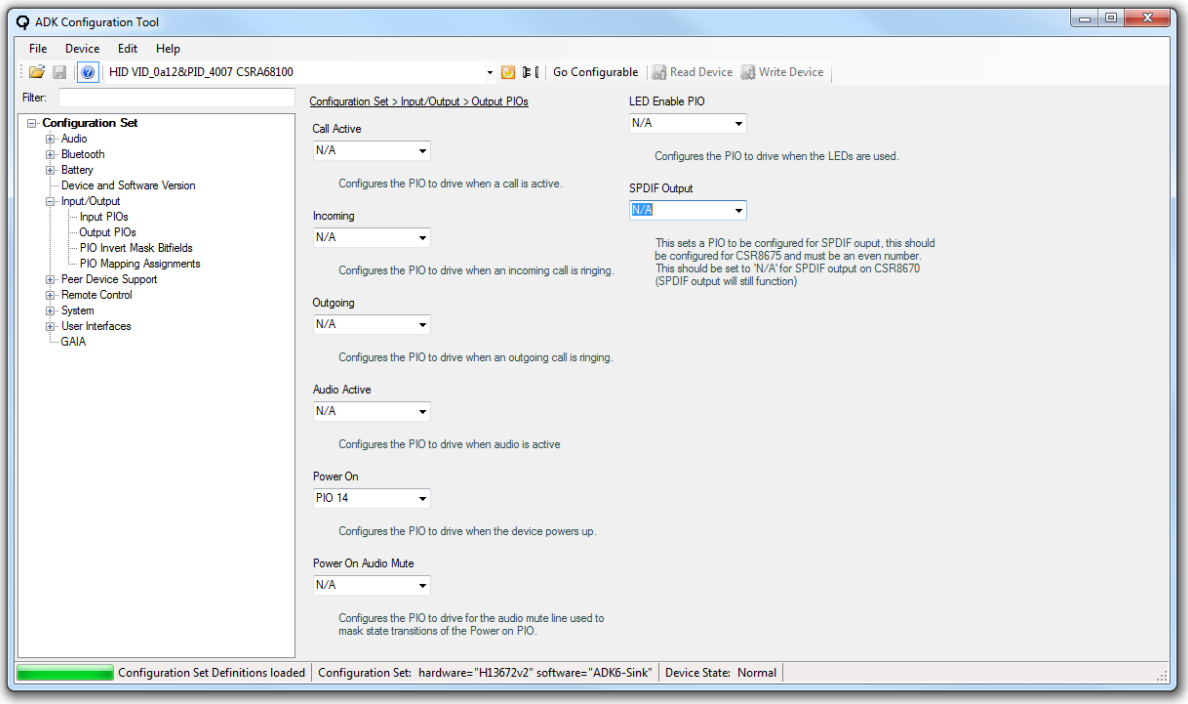


Figure 4-5 Output PIO configuration

Power up and power down operations for an external amplifier can be managed using PIOs. A Power pin may be available on the external amplifier which provides a hardware shut down option when required, for example, when the system goes into Stand-by mode.

A Mute pin may also be available on some external amplifiers. For the Soundbar application, the Mute pin is also driven by the **Audio Mute** PIO before the **Power On** PIO to minimize pops and clicks during power up and power down events. The **Audio Mute** PIO must be configured to **N/A** when a reference external amplifier board, that is H13117, is in use.

Table 4-1 shows the sequence of PIO activity when **Audio Mute** and **Power On** PIOs are configured to values other than **N/A**.

Table 4-1 Audio mute and Power on PIO operation

Action	Sequence of PIO Activity		
	1 - Audio Mute PIO	2 - Power On PIO	3 - Audio Mute PIO
Power Off (Go into Standby Mode)	Goes High	Goes Low	Goes Low
Power On	Goes High	Goes High	Goes Low

4.3 Soundbar button translation for I²S operation (ADK 4.x only)

The default Soundbar configuration requires a change to the button translation table for **Input 10**. This is because the default configuration uses **PIO5**, which is also used by the I2S_1_CLK line.

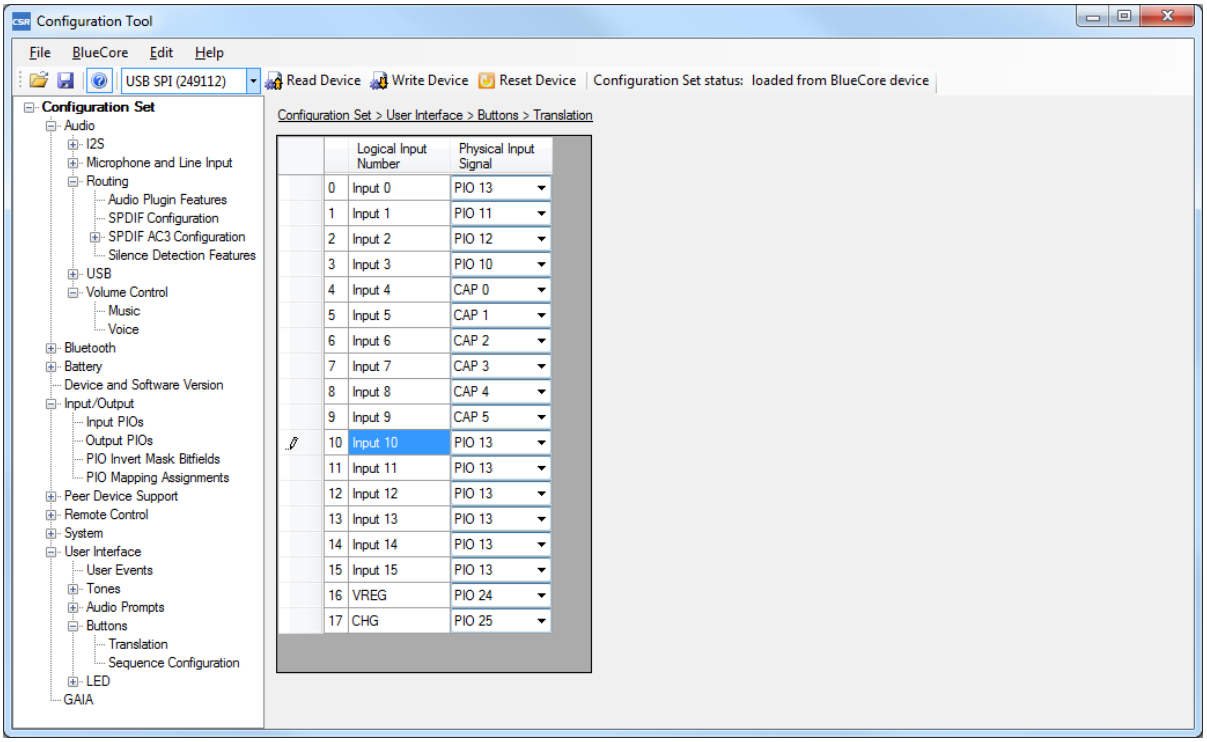


Figure 4-6 Soundbar button translation

4.4 Soundbar input select for I²S operation (ADK 4.x only)

To hear streaming audio from a connected AG on the Soundbar the input select must be changed. Press the **SENSE_3** button to choose **Select Audio Source AG1**, see Figure 4-7.

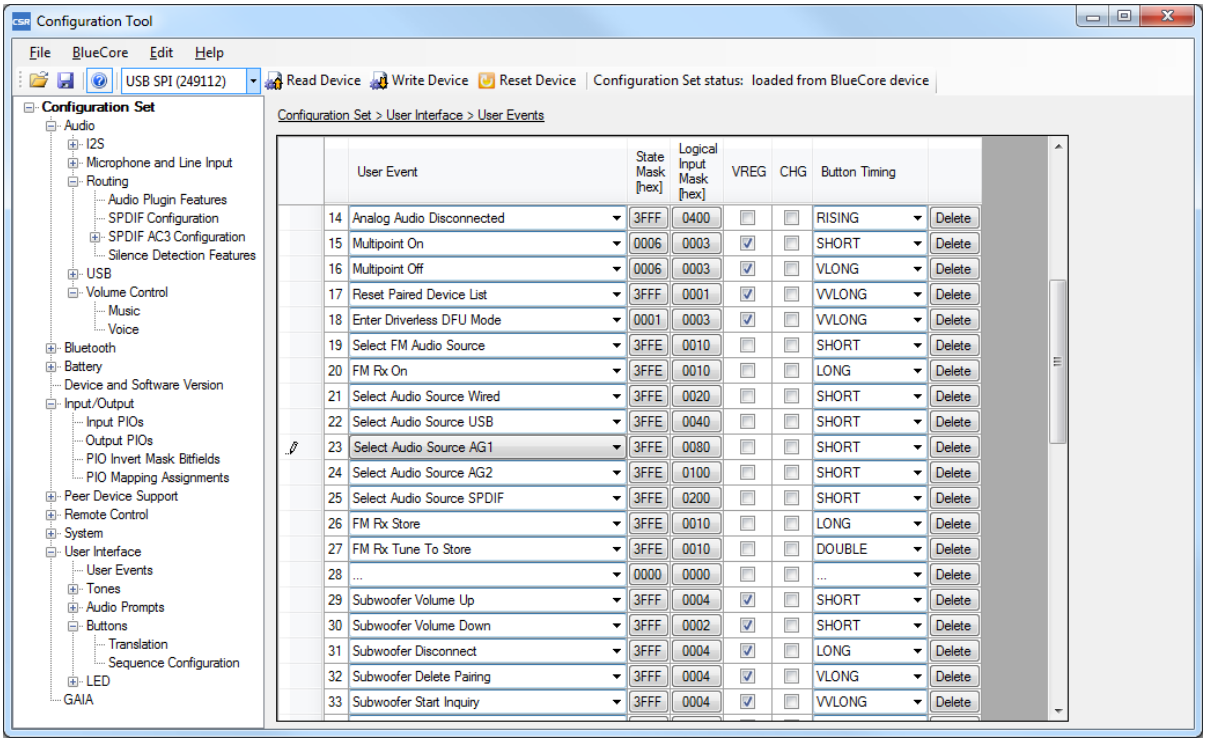


Figure 4-7 Soundbar user event configuration

4.5 I²C PIO mapping (ADK 4.x only)

These steps are not required when using ADK 6.x.

When using ADK 4.x, PSTool must be used to configure the **I2C SCL PIO Mapping** and the **I2C SDA PIO Mapping**, see Figure 4-8 and Figure 4-9.

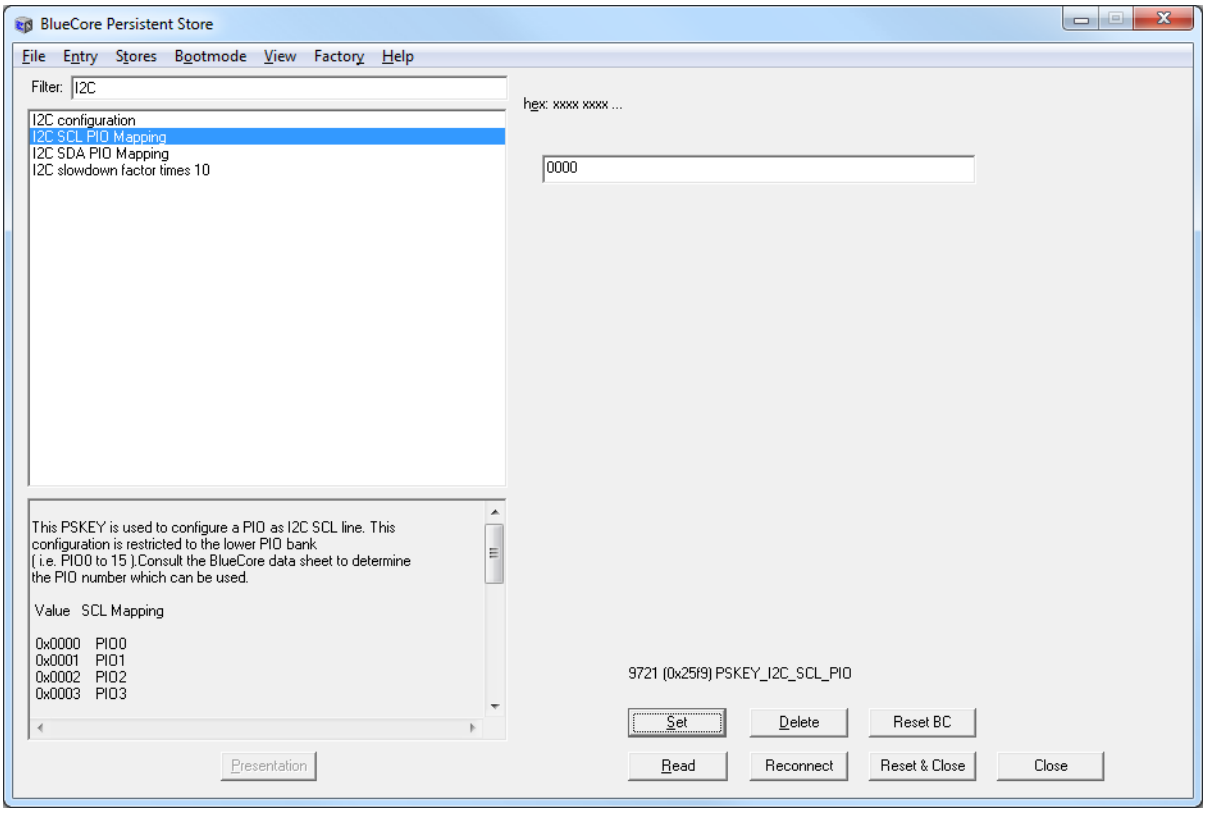


Figure 4-8 I²C SCL PIO mapping

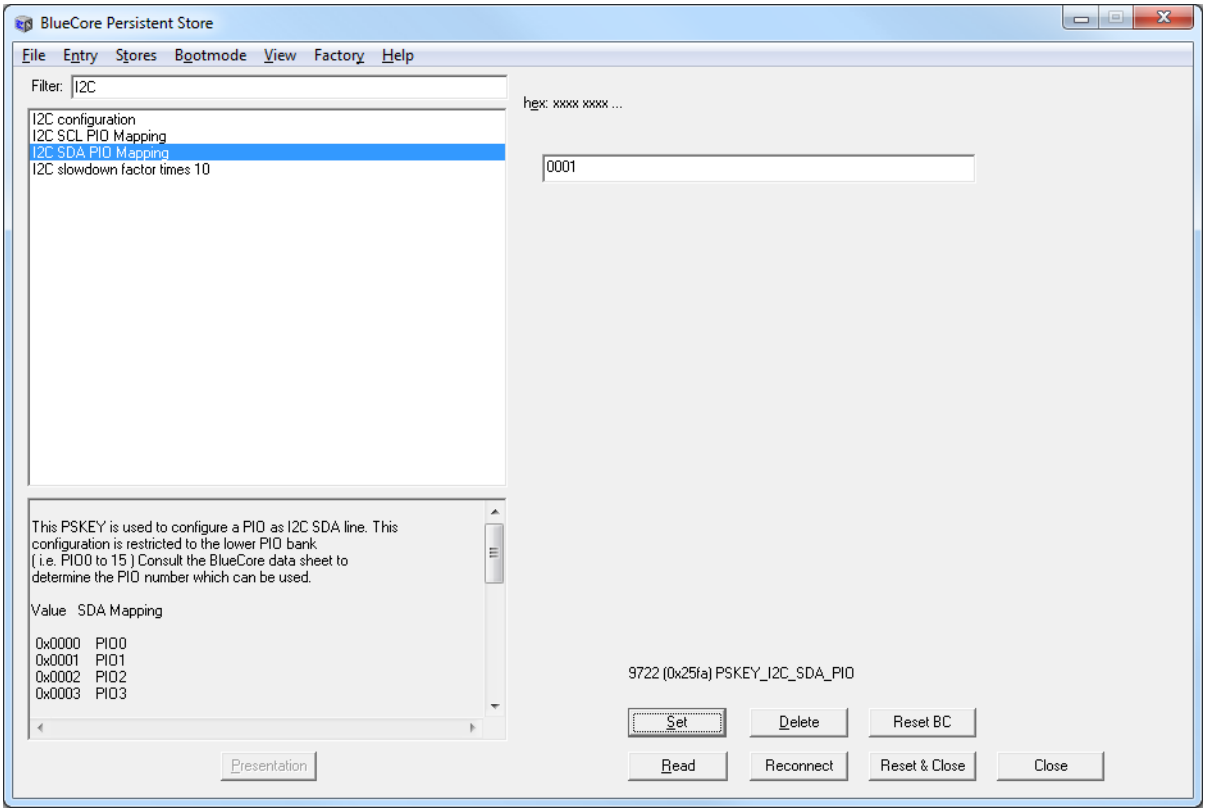


Figure 4-9 I²C SDA PIO mapping

4.6 Tone playback with I²S operation

QTIL recommends that resampling is always used with I²S. This can be enabled by modifying **Configuration Set > Audio > Routing > Audio Plugin Features** to enable **Force Re-sampling of Tones**, see Figure 4-10

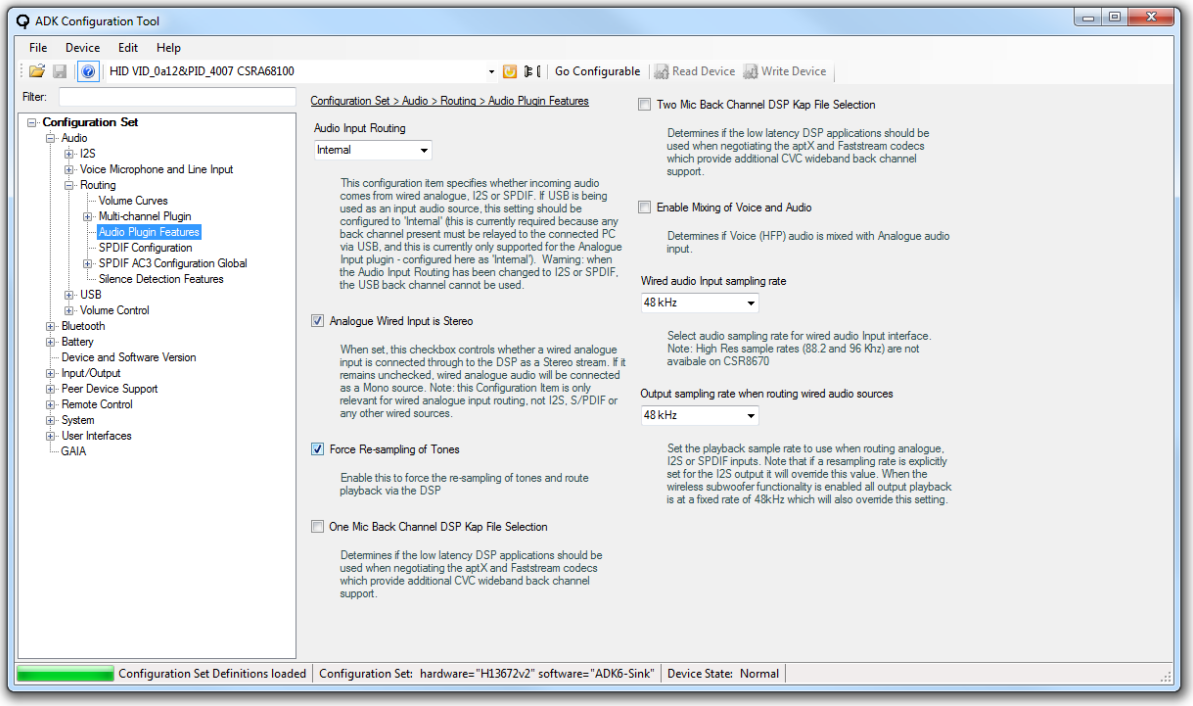


Figure 4-10 Configuration of tone playback

A I²S implementation

Capabilities of I²S implementation

I²S Master and Slave operation is supported.

Limitations of I²S implementation

The QTIL I²S implementation cannot generate a separate master clock. However, the bit clock scaling can be used to provide a faster bit clock, which can be used as a master clock. The bit clock is generated by dividing down the 48 MHz chip clock so cannot be guaranteed to have a constant mark/space ratio.

Document references

Document	Reference
Analog Devices SSM2518 Data Sheet	http://www.analog.com
ADK Audio Sink Application User Guide	80-CT439-1/CS-00236868-UG

Terms and definitions

Term	Definition
A2DP	Advanced Audio Distribution Profile
AD	Analog Devices
ADK	Audio or Application Development Kit
Bluetooth	Set of technologies providing audio and data transfer over short-range radio connections
cVc	Clear Voice Clarity
I^2C	Inter-Integrated Circuit (multimaster serial single-ended computer bus)
I^2S	Integrated Interchip Sound
ID	IDentifier
MSB	Most Significant Bit
PIO	Programmable Input/Output
PS	Persistent Store
QTIL	Qualcomm Technologies International, Ltd.
ROM	Read Only Memory
SCO	Synchronous Connection-Oriented
SDA	Serial Data (line)
USB	Universal Serial Bus