User's Guide
SLAU285–July 2009

TLV320AIC3111 EVM

This user's guide describes the operation of the TLV320AIC3111 EVM evaluation module (EVM). The EVM features a TLV320AIC3111 stereo audio codec, amplifiers for speakers and headphones, and a digital signal processing module. Together with the USB-MODEVM board, the TLV320AIC3111 Control Software and a PC running Windows™ XP it is a plug-and-play solution to evaluate the capabilities of the TLV320AIC3111.

The information in a caution or a warning is provided for your protection. Read each caution and warning carefully.

Contents

List of Figures

List of Tables

1 EVM Overview

1.1 Introduction

The TLV320AIC3111 EVM features a TLV320AIC3111 stereo audio codec, amplifiers for speakers and headphones and a digital signal processing module.

Together with the USB-MODEVM board, the TLV320AIC3111 Control Software and a PC running Windows XP, it is a plug-and-play solution to evaluate the capabilities of the TLV320AIC3111.

The USB-MODEVM board contains a TAS1020B streaming audio USB controller, which enumerates as a USB audio class device.

When the USB-MODEVM + TLV320AIC3111 EVM is connected to a PC running Microsoft Windows XP, it will be recognized as a sound card. Once the TLV320AIC3111 is configured using the TLV320AIC3111 control software, any audio playback and record software on the PC that uses the Windows audio subsystem (sound card) can use the TLV320AIC3111.

1.2 Box Contents

The following items ship with the TLV320AIC3111 EVM:

- TLV320AIC3111 EVM
- USB-MODEVM

The control software required to operate the EVM is available from the TLV320AIC3111 product folder at http://www.ti.com

1.3 Related Documentation From Texas Instruments

TLV320AIC3111 data sheet (SLAS644)

2 EVM + PC

This chapter explains how to use the TLV320AIC3111 EVM with a PC running Windows XP.

2.1 EVM Preparation

To interface the TLV320AIC3111 EVM with a PC using USB, plug the TLV320AIC3111 EVM onto the USB-MODEVM as shown in Figure 2 1. TLV320AIC3111 EVM + USB MODEVM.

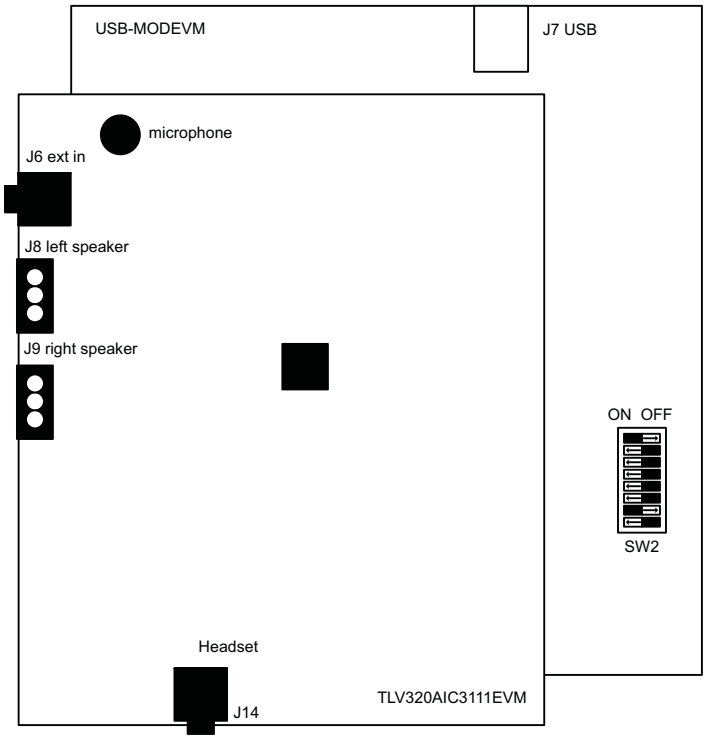


Figure 1. TLV320AIC3111 EVM + USB MODEVM

Note: USB-MODEVM configuration

To control the TLV320AIC3111 from the PC via the USB-MODEVM, set switch SW2 position 1, 3, 4, 5, 6, 7 to ON and position 2 and 8 to OFF.

2.1.1 Analog Signal Connections

- Connect a headphone to J14 (3,5 mm jack)
- Connect 8-Ω speakers to J8 and J9 (two 3-screw terminals)
  - Left speaker to SPLN and SPLP
  - Right speaker to SPRN and SPRP
- By default, the on-board microphone is connected to the ADC. To use the line input (J7), refer to sheet 1 of the AIC3111_RHB_EVM schematics (chapter 3.2)

2.2 Control Software

The TLV320AIC3111 control software exposes most features of the TLV320AIC3111 through an intuitive graphical user interface.

Note: Before Windows on the PC can use the TLV320AIC3111 EVM as a sound card, the TLV320AIC3111 on the EVM must be configured (sampling rate, audio routing, internal amplifier settings etc.) with the TLV320AIC3111 control software.

2.3 Installation

Download the TLV320AIC3111 control software (SLAC289) from the TLV320AIC3111 product folder at http://www.ti.com and launch the program (SLAC289).

This file is a self-extracting archive. The default target folder is:

C:\Program Files\Texas Instruments\AIC3111

Click the Unzip button to complete the installation.

The TLV320AIC3111 control software is now available in the target folder. The name of the executable is CodecControl.exe

To launch the TLV320AIC3111 control software, navigate to the target folder with the Windows Explorer and double click CodecControl.exe.

2.4 Concepts

The TLV320AIC3111 control software presents a block diagram view of the TLV320AIC3111 (or select modules within the TLV320AIC3111).

The block diagram consists of active objects that can react to user input (for example switches or amplifiers with variable gain that show a volume control on a mouse click event).

Note: Each active object will change color to red if the mouse cursor is above the object. Clicking the object will trigger its function.

Some active objects are linked to control register(s) of the TLV320AIC3111 in a two way fashion. If an EVM is connected, the control software will update the appropriate register(s) whenever an active object is triggered. If a register that is linked to an active object is changed via other components (for example the script interpreter or the register inspector), the active object will change its state accordingly.

The control software will automatically detect a TLV320AIC3111 EVM once it is connected to a USB port of the PC.

If no TLV320AIC3111 EVM is connected to the PC, the control software changes to a simulation mode, where it is possible to retrieve script commands based on user input within the block diagram.

2.5 Main Window

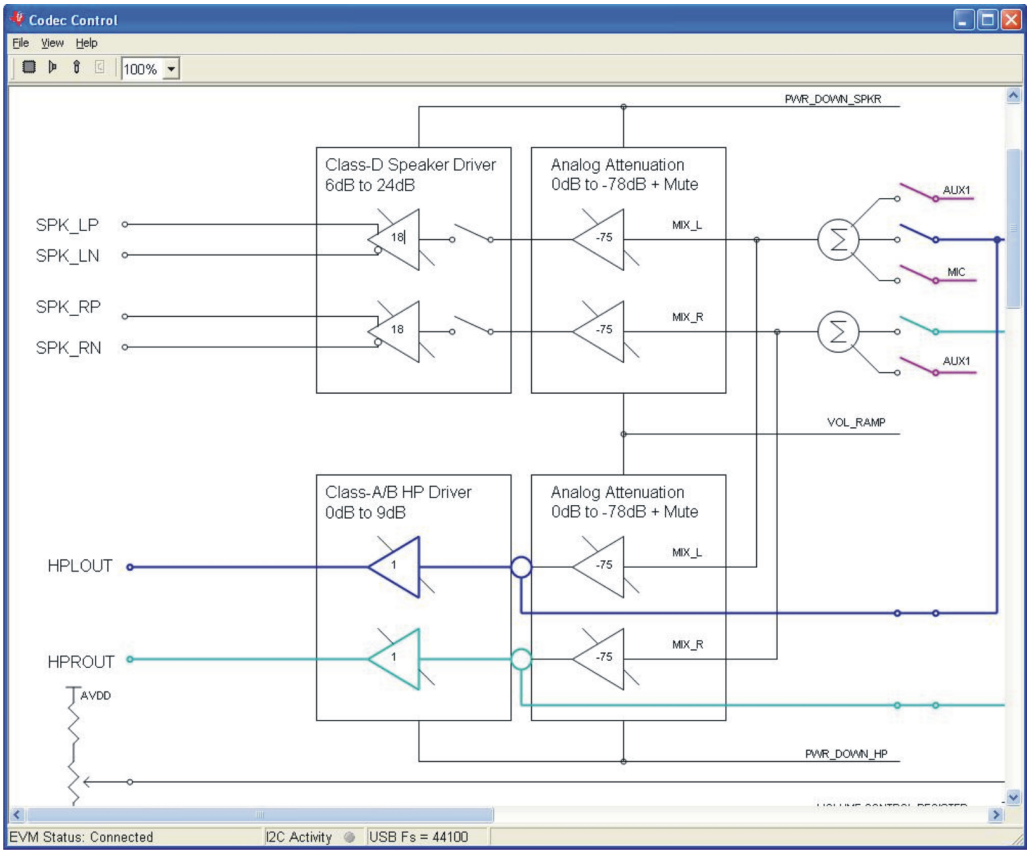


Figure 2. Main Window

At the top of the main window is a tool bar with buttons to change between four different use cases of the TLV320AIC3111:

- Full featured TLV320AIC3111
- Playback only
- Record only

By default, the control software displays the full featured block diagram of the TLV320AIC3111.

Each use case has its own initialization script, which will run if a use case is selected by clicking on one of the use case buttons. The initialization script contains register settings for the TLV320AIC3111 to configure the device for a specific use case.

The toolbar contains a control that determines the zoom factor. Change the zoom by selecting the desired zoom factor.

To move the block diagram, click on a blank area within the block diagram and drag the diagram with the mouse.

At the bottom of the main window is a status bar that provides information about the state of the communication between the control software and the TLV320AIC3111 EVM. It also shows hints about elements in the block diagram, for example the I2C page and register / bit location of a selected switch.

Audio signal paths (both digital and analog) will change color from black to

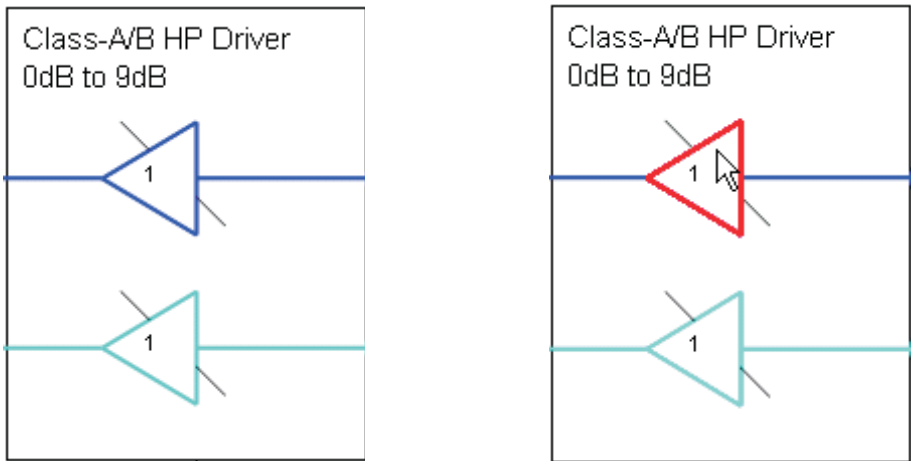
- Blue for left audio output
- Turquoise for right audio output
- Magenta for audio input

once they are activated via switches. This feature visualizes all audio paths and immediately highlights if a path is disabled.

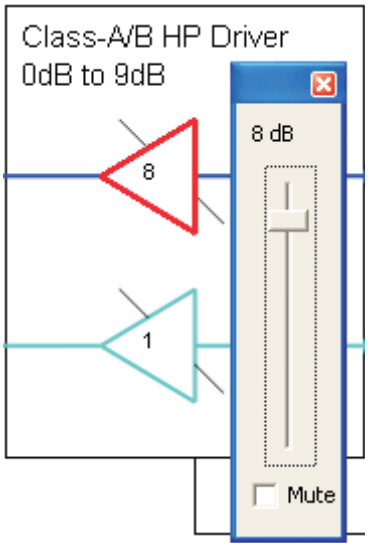
2.5.1 Using Active Objects

Moving the mouse pointer over an active object will light up the active object (the color of the object turns red).

For example, the Class-A/B HP Driver left amplifier active object will turn from its inactive state to its active state when the mouse pointer enters the amplifier symbol:



Clicking the activated object will trigger its function. In the case of the amplifier active object, the function is a volume control. Moving the volume control slider changes the volume setting of the amplifier (it is also possible to change the volume by clicking onto the number within the amplifier symbol and typing the new gain setting).The control software updates the appropriate register in the TLV320AIC3111 and as a result the volume on the headphone output will change accordingly.



2.6 Dialogs and Active Objects

The TLV320AIC3111 control software contains several dialog windows that give access to additional features.

Most dialogs are linked to active objects and are opened by clicking on the active object.

A few dialogs are not linked to active objects and are opened using the View menu.

2.6.1 Init Script Dialog

Each use case , , , owns a unique initialization script which will automatically run when a TLV320AIC3111 EVM is detected or if the user selects another use case.

To show or edit the initialization script, choose View->Init Script… from the main window menu bar.

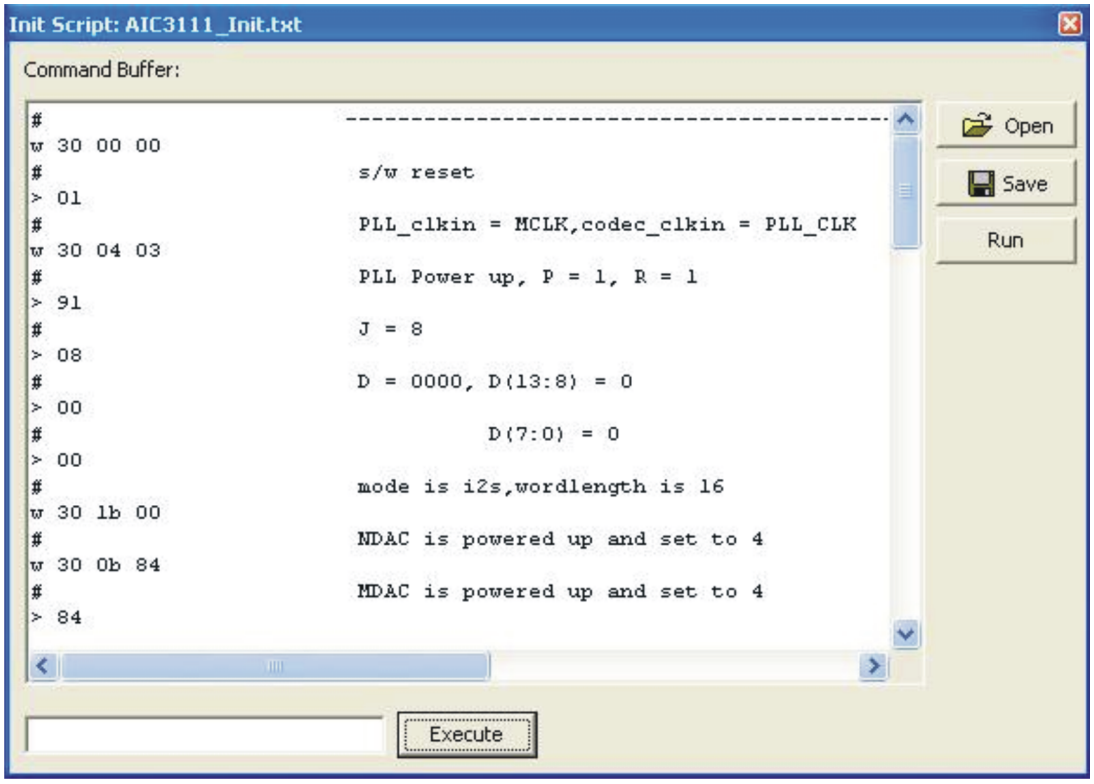


Figure 3. Initialization Script

Click the Run button to run the script again. For further information about the script syntax, see Figure 3.

2.6.2 Command Dialog

Open the command dialog (View->Command…) to write, edit, load, save and run command scripts. Command scripts are text files that contain commands to communicate with the TLV320AIC3111. The syntax is described in Figure 4.

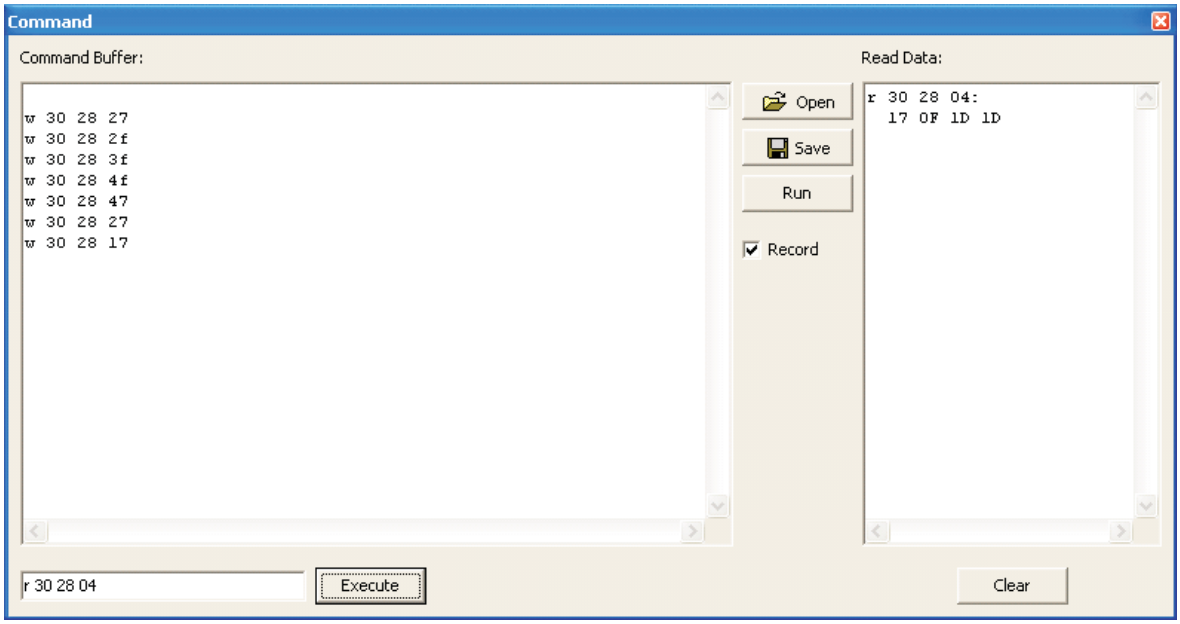


Figure 4. Command Dialog

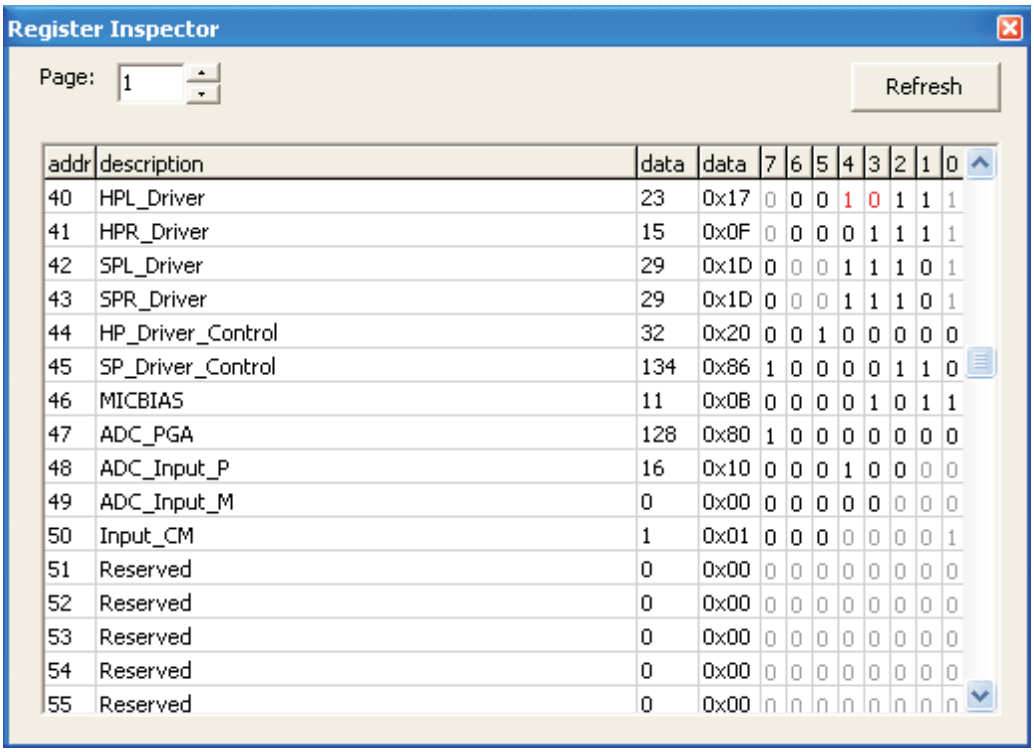
- The main area of the command dialog is command buffer (editable text) which contains the command script. To run the command script, click the Run button.
- The smaller read only text area on the right side of the command dialog displays control data read from the TLV320AIC3111. The Clear button clears the Read Data field.
- The one line text edit field on the left bottom allows single command execution.
- The Record check box enables recording of commands generated by the control software.

Figure 4 shows a recording of the volume control for the left Class-A/B HP Driver amplifier (note that the Record checkbox is checked).

A single command to read four bytes starting at address 0x28 was executed and the result is displayed in the Read Data field.

2.6.3 Register Inspector

The register inspector dialog (View->Register Inspector…) gives access to all registers of the TLV320AIC3111.



addr	description	data	data	7	6	5	4	3	2	1	0
40	HPL_Driver	23	0x17	0	0	0	1	0	1	1	1
41	HPR_Driver	15	0x0F	0	0	0	0	1	1	1	1
42	SPL_Driver	29	0x1D	0	0	0	1	1	1	0	1
43	SPR_Driver	29	0x1D	0	0	0	1	1	1	0	1
44	HP_Driver_Control	32	0x20	0	0	1	0	0	0	0	0
45	SP_Driver_Control	134	0x86	1	0	0	0	0	1	1	0
46	MICBIAS	11	0x0B	0	0	0	0	1	0	1	1
47	ADC_PGA	128	0x80	1	0	0	0	0	0	0	0
48	ADC_Input_P	16	0x10	0	0	0	1	0	0	0	0
49	ADC_Input_M	0	0x00	0	0	0	0	0	0	0	0
50	Input_CM	1	0x01	0	0	0	0	0	0	0	1
51	Reserved	0	0x00	0	0	0	0	0	0	0	0
52	Reserved	0	0x00	0	0	0	0	0	0	0	0
53	Reserved	0	0x00	0	0	0	0	0	0	0	0
54	Reserved	0	0x00	0	0	0	0	0	0	0	0
55	Reserved	0	0x00	0	0	0	0	0	0	0	0

Figure 5. Register Inspector

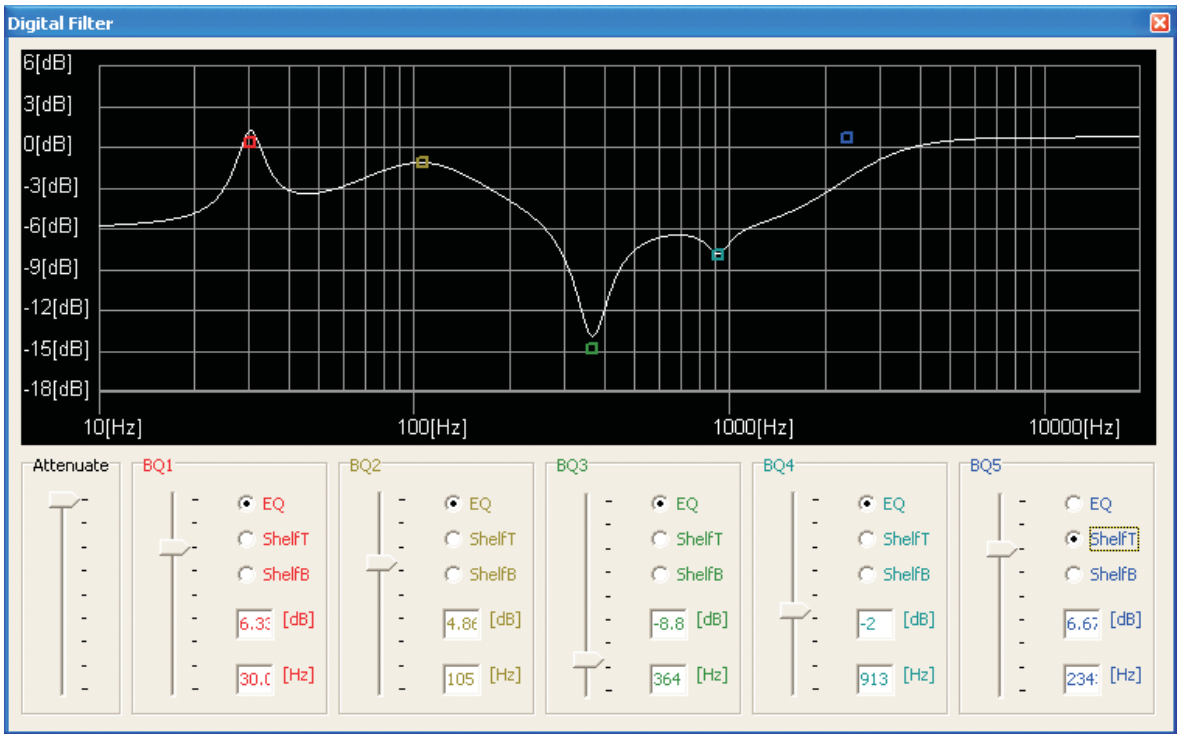
The register inspector displays the content of the TLV320AIC3111 registers. The control software will read all TLV320AIC3111 registers when a TLV320AIC3111 EVM is detected. To force reading the content of one page, click the Refresh button.

- The Page edit field selects the page to be displayed.
- The addr column shows the address of the registers within the selected page in decimal notation.
- The description column contains a description for each register. If the register has no function assigned, it is declared Reserved.
- The data columns show the data of each register (one byte). The first data column uses decimal notation, the second uses hexadecimal notation. It is possible to change the register value by clicking into one of the data fields and typing the new value (either decimal or hexadecimal).
- The numbered columns show the register content in binary notation. Read/Write bits are shown solid black or red; read only bits are gray or dark red. Red numbers represent bits that recently changed. To change a single writeable bit, click on the bit and it will flip.

2.6.4 DAC Filter

One of the digital signal processing blocks of the TLV320AIC3111 implements five digital biquad filters. The DAC digital filter dialog (View->DAC Filter…) allows real time graphical manipulation of the digital filters.

The control software will automatically configure the digital signal processing block when the DAC digital filter dialog is opened.



The digital filter dialog limits the range of each digital biquad filter to +/-12[dB] (this is an arbitrary limitation for demonstration purposes).

- Each biquad has its own unique handle with a unique color. Each handle will light up white if the mouse pointer is in the vicinity, showing that it can be selected. To change the frequency and gain of a biquad, grab and drag its handle.

  It is also possible to change the gain using the slider for each biquad.
- Each biquad can be configured for parametric EQ, Shelf Treble or Shelf Bass. If it is configured for EQ, press the shift key before selecting the handle to adjust the bandwidth of the EQ using the mouse pointer.
- Due to digital range limitations, the biquads will automatically scale, if the biquad coefficients exceed the limitations.

  The coordinate system will shift accordingly to reflect the resulting attenuation.
- To avoid clipping, add additional attenuation with the Attenuate slider.
- To retrieve the biquad coefficients, open the command dialog (see 0) and check Record.

2.6.5 Clock and Digital Signal Routing

The TLV320AIC3111 has a flexible and complex clock and digital signal routing architecture.

Two processors can connect to the TLV320AIC3111 using two separate I2S™ interfaces: The primary I2S interface has dedicated pins whereas the secondary I2S interface signals can be assigned to a selection of pins.

The TLV320AIC3111 has an on-chip clock generation module which can be configured to generate the sampling rate, modulator clocks, converter clocks, bit clock and word clock.

Click on the “Digital Audio Processing Serial Interface” active object (if it is not within the current scope of the main window, drag the block diagram to the left until the active object appears). This will change the block diagram to the clock and digital signal routing diagram:

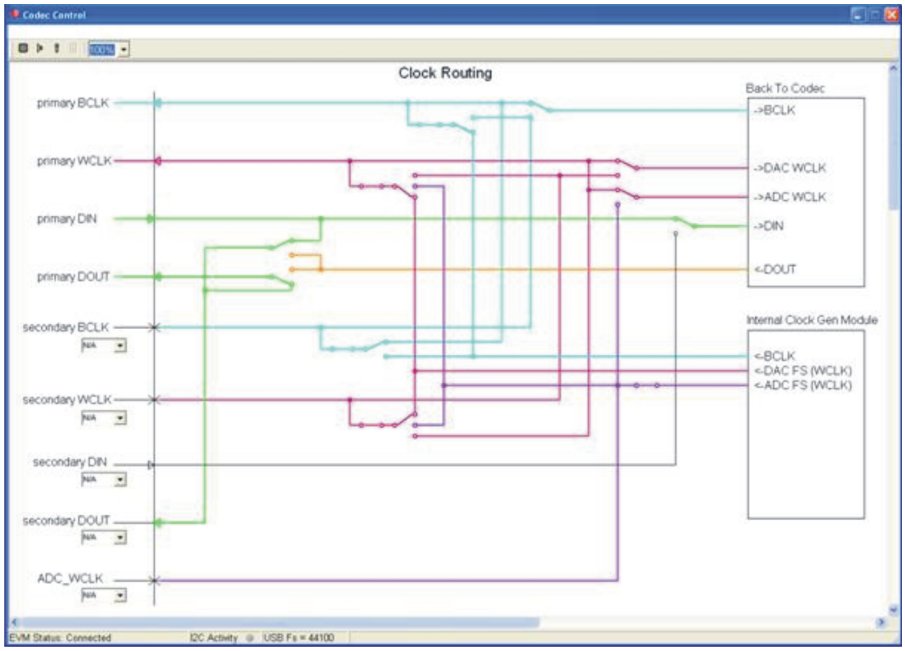


Figure 6. Clock and Digital Signal Routing

The clock and digital signal routing diagram shows the current state of the TLV320AIC3111 routing configuration and allows interactive manipulation.

- Each clock or signal source has its own unique color. For example, the BCLK signal from the internal clock generation module has a turquoise color.
- To trace the routing of a specific signal, follow its color. The example in Figure 6 shows that the BCLK signal from the internal clock generation module is routed to the primary I2C™ BCLK pin (which is configured as an output), to the secondary I2S BCLK signal (which is not connected to a pin) and to the BCLK input of the codec (ADC and DAC within the TLV320AIC3111).
- To change the definition of a pin (input or output), click the active object (arrow) that belongs to the pin. Only pins that can change between input and output are linked to such an active object. The clock routing diagram will automatically change to reflect the new routing.
- Some of the switches within the diagram are active objects, which can be manipulated using the mouse pointer. Other switches open or close depending on the state of the associated pin.
- To assign a pin to a signal of the secondary I2S interface, choose one of the available pins in the drop down box that belongs to the signal. The list of available pins will change automatically depending on the assignment of other signals to pins.
- Click on the “Back To Codec” active object to return to the previous block diagram.
- Click on the “Internal Clock Gen Module” active object to display the digital configuration dialog.

2.6.6 Digital Configuration

The digital configuration dialog gives access to the codec clock and PLL settings as well as the audio interface settings.

To open the digital configuration dialog, navigate to the clock and digital signal routing diagram (see Figure 6) and click on the “Internal Clock Gen Module” active object.

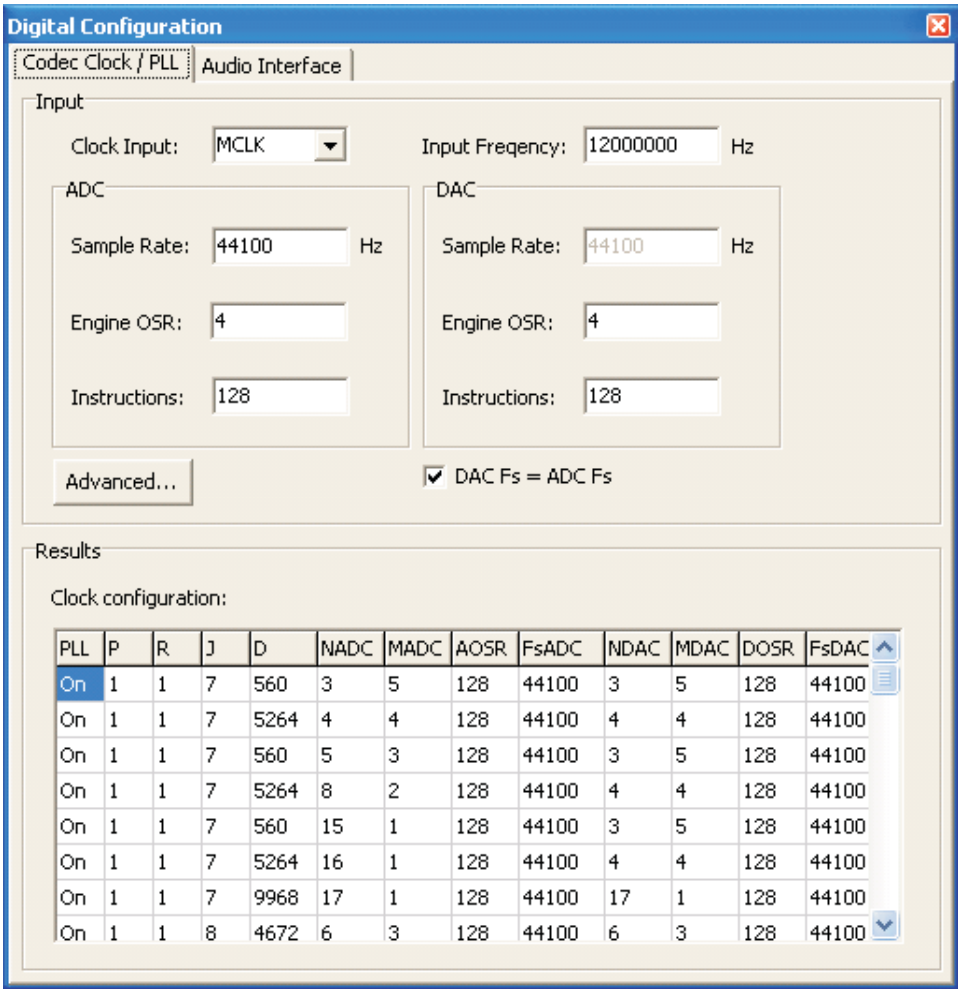


Figure 7. Digital Configuration: Codec Clock / PLL

The digital configuration dialog contains two tabs, one for the Codec Clock / PLL settings and one for the Audio Interface settings.

The Codec Clock / PLL settings tab (see Figure 7) enables simple generation of PLL and clock divider settings based on the available input frequency and the desired sample rate:

1. Choose the clock input using the Clock Input drop down box.
2. Type the available input frequency in the Input Frequency edit field.
3. Type the desired sample rate in the Sample Rate edit field of the ADC. By default, the DAC sample rate equals the ADC sample rate. Uncheck DAC Fs = ADC Fs and enter the DAC sample rate for different sample rates.
4. The Engine OSR and Instructions fields affect the miniDSP. Please contact your TI representative for further information about the miniDSP.
5. The Results list shows all clock settings that fulfill the chosen parameters. Double click on one of the results to program the TLV320AIC3111 with the new settings.

Each result has the following columns:

- PLL: On or Off
- P,R,J,D: PLL configuration
- NADC, MADC: ADC clock dividers
- AOSR: ADC over-sampling factor
- FsADC: ADC sampling rate
- NDAC, MDAC: DAC clock dividers
- DOSR: DAC over-sampling factor
- FsDAC: DAC sampling rate

Click the Advanced… button to show the advanced clock settings dialog.

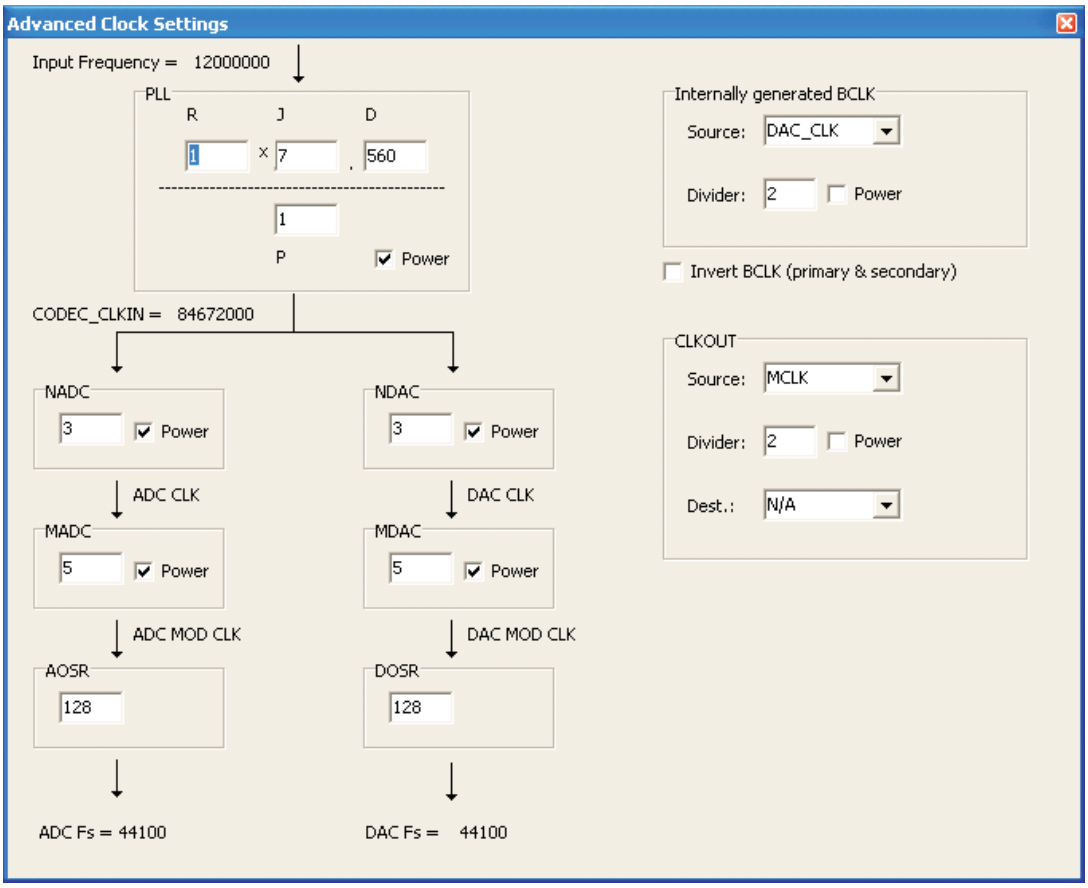


Figure 8. Advanced Clock Settings.

The advanced clock settings dialog gives direct access to the PLL and codec clock dividers. It will recalculate the clock results dynamically whenever a parameter is changed.

The internally generated bit clock signal (BCLK) can be derived from several sources and divided by an integer number. Select the desired source with the Source drop down box, choose the divisor and enable power to the divider, if required.

It is possible to put out a clock signal CLKOUT. Select the clock source, the divider and the destination pin using the advanced clock settings dialog.

2.6.7 Audio Interface

The Audio Interface tab (see Figure 9) contains controls to manipulate the digital audio interface:

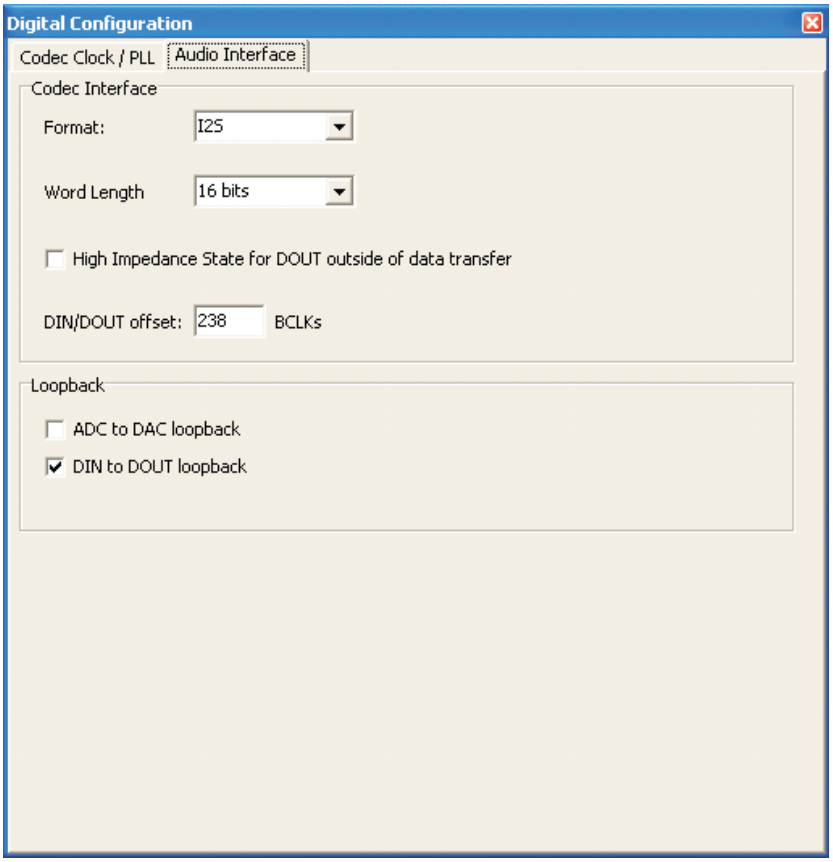


Figure 9. Audio Interface

Use the Format drop down box to change the digital audio interface format:

- I2S
- DSP
- Right Justified
- Left Justified

For details about the digital audio interface formats see the TLV320AIC3111 data sheet (SLAS550), 5.4 AUDIO DIGITAL I/O INTERFACE.

The Word Length drop down box defines the number of bits per audio word.

The DIN/DOUT offset defines where the data for the ADC or from the DAC is located in the bit-stream. This is required for TDM (DSP) interface format.

2.6.8 AGC

The TLV320AIC3111 has an automatic gain control module, which is accessible by an active object labeled AGC within the block diagram for the full TLV320AIC3111 .

Clicking on the AGC active objects opens the AGC Dialog:

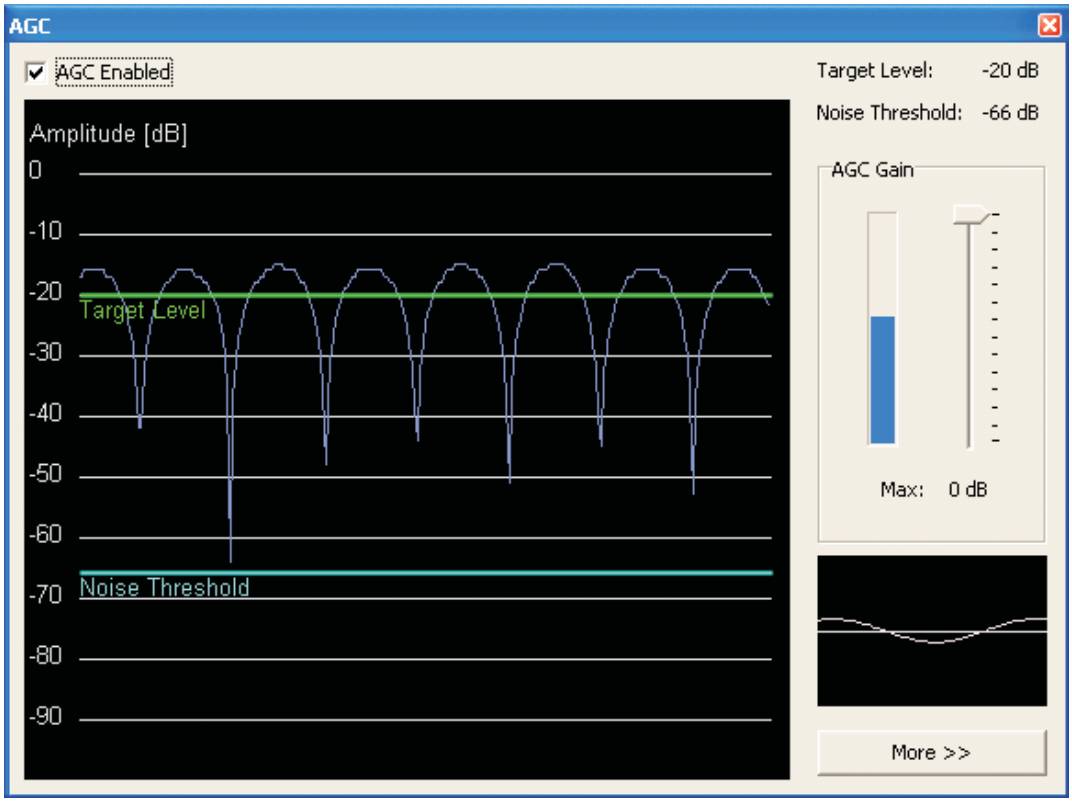


Figure 10. Automatic Gain Control

If the AGC is enabled, the TLV320AIC3111 will adjust the gain of the analog audio input signal amplifier so that the input signal level for input signal amplitudes above the noise threshold approximates the target level.

- The main display in the AGC dialog shows the Amplitude of the ADC output data in decibel with 0dB equal to a full scale signal.
-
  - The target level line can be adjusted using the mouse pointer. It will change color to red if it the mouse pointer is in the vicinity, indicating that it can be moved (click and drag).
  - The noise threshold line is also adjustable
- The small display on the left shows the ADC output data
- The AGC Gain field shows the applied gain (if the AGC is enabled) and allows setting a maximum gain using the slider.

Advanced AGC controls are available by clicking the More button. This will reveal further controls to adjust various AGC parameters.

2.6.9 Digital Volume Control and DRC

The TLV320AIC3111 has digital volume control and dynamic range compression modules for each DAC channel. Each is accessible by a active objects labeled DVol within the block diagram for the full TLV320AIC3111 and the playback use case .

Clicking on the DVol active object opens the DAC Vol dialog, which contains a slider to set the digital volume and several options. Checking the DRC option reveals the DRC transfer function.

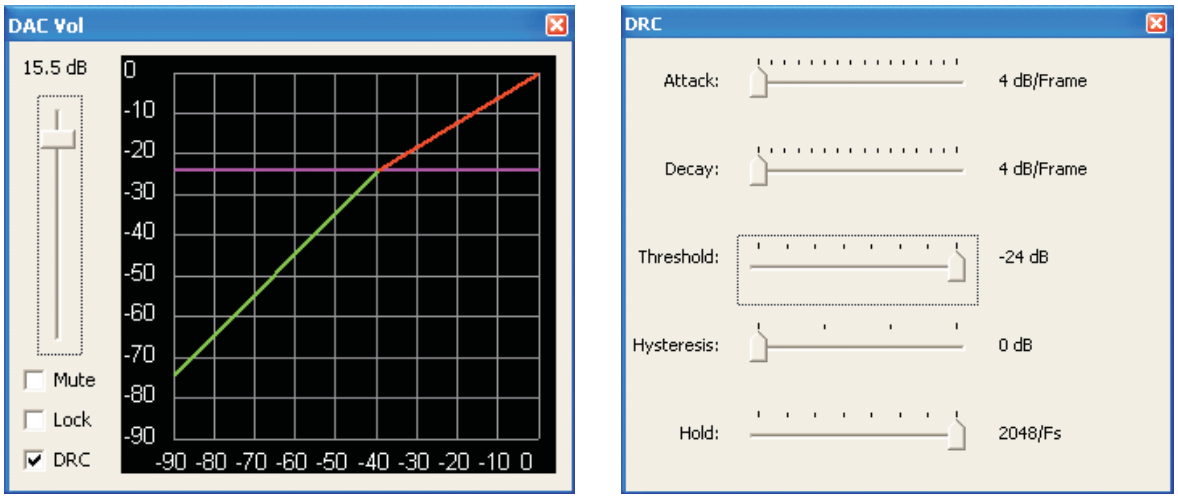


Figure 11. DRC Transfer Function and DRC Dialog

The horizontal axis of the DRC transfer function shows the input to the DRC and the vertical axis shows the output of the DRC. The green line shows the gain below the DRC threshold, the magenta colored horizontal line shows the DRC threshold and the red line shows the gain above the DRC threshold.

The DRC transfer function will change depending on the digital volume setting and the DRC threshold.

Click on the DRC transfer function to reveal the DRC dialog, which contains a slider to change the DRC threshold.

3 EVM Hardware

This chapter contains information about the EVM Hardware (switches, jumpers, schematics).

3.1 Connectors and Jumpers

Table 1. Analog I/O

Connector		Function
J6	1	MIC1LP
	2	AGND
	3	MIC1LM or MIC1RP
J7		Microphone
J8	1	SPLM (Speaker)
	2	AGND
	3	SPLP (Speaker)
J9	1	SPRM (Speaker)
	2	AGND
	3	SPRP (Speaker)
J10	1	HPL
	2	AGND
	3	HPR
J11	1	SPRM (Speaker Filtered)
	2	AGND
	3	SPRP (Speaker Filtered)
J12	1	BATT.SVDD
	2	AGND
J13	1	HPL (Headphone)
	2	AGND
	3	HP (Headphone)
J14		Headset
J15		Headphone Filtered

Table 2. Jumpers

Jumper	Function	Positions	Default
W1	MIC bias select	1-2: 3.3V	1-2: 3.3V
		2-3: EVM	
W2, W3	On-board MIC	1-2: add MIC	populated
W4	MIC bias to MIC1RP	1-2: connect	populated
W5	MIC bias to MIC1LP	1-2:connect	populated
W6	MIC bias to MIC1RP load	1-2: 1.0k	populated
		Removed: 2.2k	
W7	J7 MIC bias select	1-2: MIC1RP	1-2: MIC1RP
		2-3: MIC1LM	
W8	MIC1LM termination	1-2: AC to AGND	1-2: AC to AGND
		2-3: 1.0k to AGND	
W9	VOL/HED_DET select	1-2: VOL	1-2: VOL
		2-3: HED_DET	
W10	AVDD current wire loop	1-2: connected	populated (remove to measure current)

W11	SVDD current wire loop	1-2: connected	populated (remove to measure current)
W12	SVDD current wire loop	1-2: connected	populated (remove to measure current)
W13	HVDD current wire loop	1-2: connected	populated (remove to measure current)
W14	DVDD current wire loop	1-2: connected	populated (remove to measure current)
W15	IOVDD current wire loop	1-2: connect	populated (remove to measure current)

Table 3. Expansion Connectors P4/J4

Pin Number	Signal	Description
P4.1/J4.1	NC	
P4.2/J4.2	NC	
P4.3/J4.3	NC	
P4.4/J4.4	DGND	Digital Ground
P4.5/J4.5	NC	
P4.6/J4.6	NC	
P4.7/J4.7	NC	
P4.8/J4.8	NC	
P4.9/J4.9	NC	
P4.10/J4.10	DGND	Digital Ground
P4.11/J4.11	NC	
P4.12/J4.12	NC	
P4.13/J4.13	NC	
P4.14/J4.14	$\overline{\text{RESET}}$	TAS1020B Reset
P4.15/J4.15	NC	
P4.16/J4.16	NC	
P4.17/J4.17	NC	
P4.18/J4.18	DGND	Digital Ground
P4.19/J4.19	NC	
P4.20/J4.20	NC	

Table 4. Expansion Connectors P5/J5

Pin Number	Signal	Description
P5.1/J5.1	NC	
P5.2/J5.2	NC	
P5.3/J5.3	BCLK	Audio Serial Data Bus Bit Clock
P5.4/J5.4	DGND	Digital Ground
P5.5/J5.5	NC	
P5.6/J5.6	NC	
P5.7/J5.7	WCLK	Audio Serial Data Bus Word Clock
P5.8/J5.8	NC	
P5.9/J5.9	NC	
P5.10/J5.10	DGND	Digital Ground
P5.11/J5.11	SDIN	Audio Serial Data Bus Data Input
P5.12/J5.12	NC	
P5.13/J5.13	SDOUT	Audio Serial Data Bus Data Output
P5.14/J5.14	NC	

P5.15/J5.15	NC	
P5.16/J5.16	SCL	I2C Clock
P5.17/J5.17	MCLK	Master Clock Input
P5.18/J5.18	DGND	Digital Ground
P5.19/J5.19	NC	
P22.20/J5.20	SDA	I2C Data

Table 5. Power Supply

Pin Number	Signal
P3.1/J3.1	NC
P3.2/J3.2	NC
P3.3/J3.3	+5VA
P3.4/J3.4	NC
P3.5/J3.5	DGND
P3.6/J3.6	AGND
P3.7/J3.7	+1.8VD
P3.8/J3.8	NC
P3.9/J3.9	+3.3VD
P3.10/J3.10	NC

3.2 EVM Schematics

The schematic diagram for the TLV320AIC3111EVM is provided as a reference.

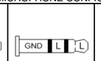
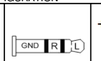
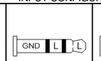
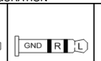
Revision History		
REV	ECN Number	Approved

Daughtercard_Interface
Daughtercard_Interface.SCH

SCHEMATIC PAGE 2

Circuit
Circuit.SCH

SCHEMATIC PAGE 3

		EXTERNAL ELECTRET MICROPHONE CONFIGURATION				JACK LINE INPUT CONFIGURATION				ONBOARD ELECTRET MICROPHONE CONFIGURATION	
											
MODE		SINGLE-ENDED MONO	SINGLE-ENDED MONO	SINGLE-ENDED STEREO	DIFFERENTIAL MONO	SINGLE-ENDED MONO	SINGLE-ENDED MONO	SINGLE-ENDED STEREO	DIFFERENTIAL MONO	SINGLE-ENDED MONO	DIFFERENTIAL MONO
CODEC INPUTS		MIC1LP	MIC1LP	MIC1LP MIC1RP	MIC1LP MIC1LM	MIC1LP	MIC1LP MIC1RP	MIC1LP MIC1RP	MIC1LP MIC1LM	MIC1LP	MIC1LP
ONBOARD MIC JUMPERS	W2	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	IN	IN
	W3	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	IN	IN
INPUT CONFIG JUMPER SETTINGS	W4	IN	IN	IN	OUT	OUT	OUT	OUT	OUT	IN	OUT
	W5	OUT	OUT	OUT	IN	OUT	OUT	OUT	OUT	OUT	IN
	W6	N/A	N/A	N/A	IN	N/A	N/A	N/A	N/A	N/A	IN
	W7	OUT	OUT	1-2	2-3	OUT	1-2	1-2	2-3	2-3	2-3
	W8	N/A	N/A	OUT	2-3	1-2	1-2	1-2	OUT	1-2	2-3
		Microphone bias provided on tip.	Microphone bias provided on tip, ring not connected to circuit.	Stereo electret microphones. Bias provided to both inputs.	Differential electret microphone.	MIC1LM is AC-coupled to AVSS.	MIC1LM is AC-coupled to AVSS, ring is connected to MIC1RP	MIC1LM is AC-coupled to AVSS, ring is connected to MIC1RP	Differential line in.	Single Ended - Mono.	Differential - Mono.

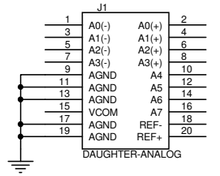
TEXAS INSTRUMENTS

DATA ACQUISITION PRODUCTS
HIGH PERFORMANCE ANALOG DIVISION
SEMICONDUCTOR GROUP

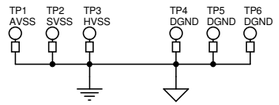
12500 T.I. Boulevard, Dallas, Texas 75243 USA

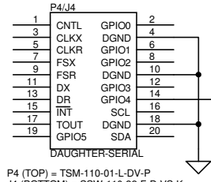
ENGINEER **Mike Tsecouras**	TITLE **AIC3111_RHB_EVM**		
DRAWN BY **Steve Leggio**			
DOCUMENT CONTROL NO **N/A**	SIZE **A**	DATE **16-Mar-2009**	REV **A**
SHEET **1** OF **3**	FILE		

REVISION HISTORY		
REV	ENGINEERING CHANGE NUMBER	APPROVED



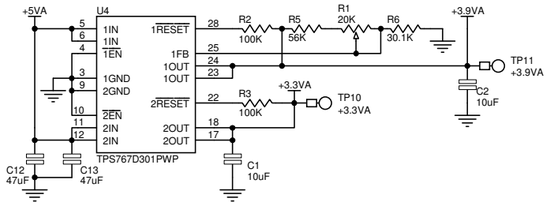
J1 (BOTTOM) = SSW-110-22-F-D-VS-K

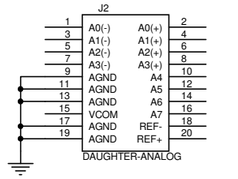




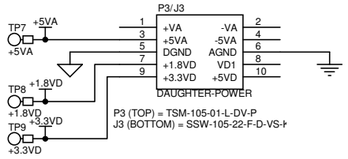
P4 (TOP) = TSM-110-01-L-DV-P
J4 (BOTTOM) = SSW-110-22-F-D-VS-K

/RESET

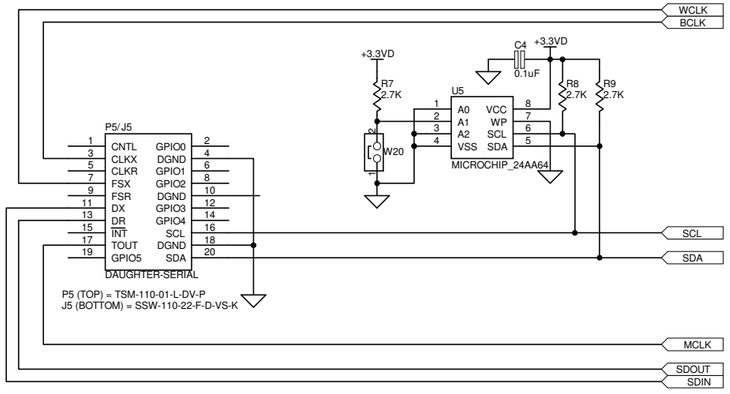




J2 (BOTTOM) = SSW-110-22-F-D-VS-K



P3 (TOP) = TSM-105-01-L-DV-P
J3 (BOTTOM) = SSW-105-22-F-D-VS-K



P5 (TOP) = TSM-110-01-L-DV-P
J5 (BOTTOM) = SSW-110-22-F-D-VS-K

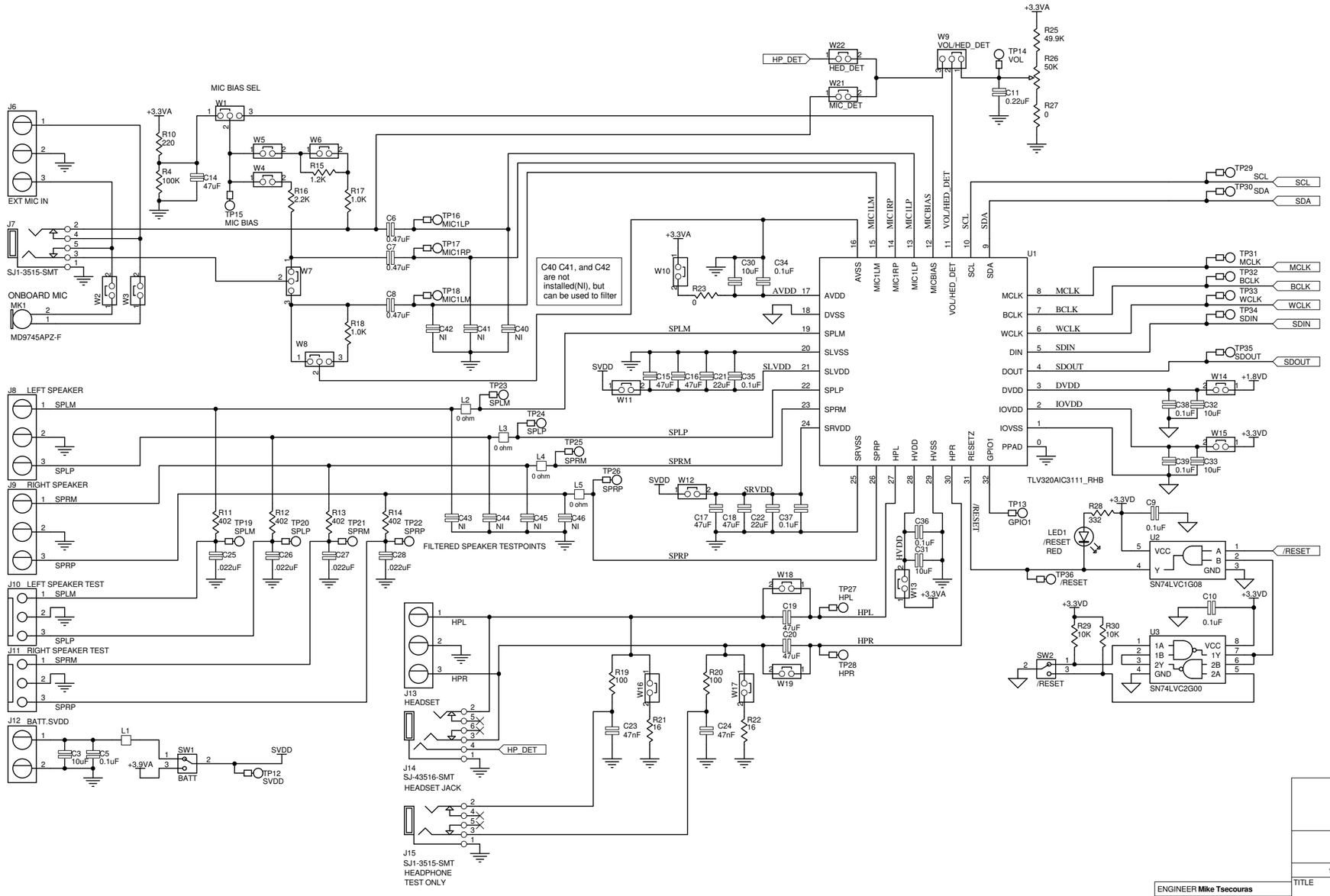
TEXAS INSTRUMENTS

DATA ACQUISITION PRODUCTS
HIGH-PERFORMANCE ANALOG DIVISION
SEMICONDUCTOR GROUP

12500 T.I. Boulevard, Dallas, Texas 75243 USA

ENGINEER **Mike Tsecouras**	TITLE **AIC3111_RHB_EVM**	
DRAWN BY **Steve Leggio**		
DOCUMENT CONTROL NO **N/A**	SIZE **B**	DATE **16-Mar-2009** REV **A**
SHEET **2** OF **3**	FILE	

Revision History		
REV	ECN Number	Approved



TEXAS INSTRUMENTS			
DATA ACQUISITION PRODUCTS HIGH PERFORMANCE ANALOG DIVISION SEMICONDUCTOR GROUP			
12500 T.I. Boulevard, Dallas, Texas 75243 USA			
ENGINEER **Mike Tsecouras**	TITLE **AIC3111_RHB_EVM**		
DRAWN BY **Steve Leggio**			
DOCUMENT CONTROL NO **N/A**	SIZE **A**	DATE **16-Mar-2009**	REV **A**
SHEET **3** OF **3**	FILE		

3.3 EVM Bill of Materials

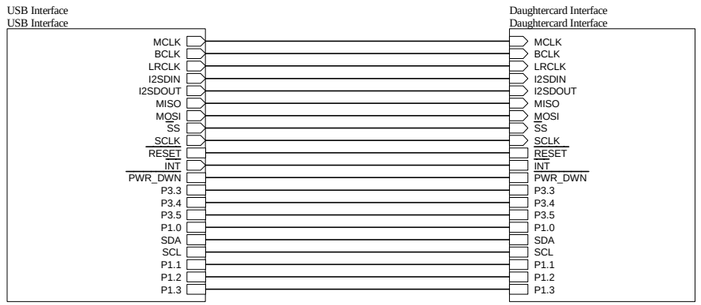
Qty	Value	Ref Des	Description
1	0	R27	RES 0 Ω 1/10W 5% 0603 SMD
1	0	R23	RES 0 Ω 1/8W 5% 0805 SMD
4	0	L2, L3, L4, L5	RES 0 Ω 1/4W 5% 1206 SMD
2	16 Ω	R21, R22	RES 16 Ω 1W 5% 2512 SMD
2	100 Ω	R19, R20	RES 100 Ω 1/10W 1% 0603 SMD
1	220 Ω	R10	RES 220 Ω 1/10W 5% 0603 SMD
1	332 Ω	R28	RES 332 Ω 1/10W 1% 0603 SMD
4	402 Ω	R11, R12, R13, R14	RES 402 Ω 1/10W 1% 0603 SMD
2	1.0 kΩ	R17, R18	RES 1.00 kΩ M 1/10W 1% 0603 SMD
1	1.2 kΩ	R15	RES 1.20 kΩ 1/10W 1% 0603 SMD
1	2.2 kΩ	R16	RES 2.2 kΩ 1/10W 5% 0603 SMD
3	2.7 kΩ	R7, R8, R9	RES 2.7 kΩ1/10W 5% 0603 SMD
2	10 kΩ	R29, R30	RES 10 kΩ 1/10W 5% 0603 SMD
1	20 kΩ	R1	TRIMPOT 20 kΩ 4MM TOP ADJ SMD
1	30.1 kΩ	R6	RES 30.1 kΩ 1/10W 1% 0603 SMD
1	49.9 kΩ	R25	RES 49.9 kΩ 1/10W 1% 0603 SMD
1	50 kΩ	R25	POT 50 kΩ 3/8" SQ CERM SL ST
1	56.0 kΩ	R5	RES 56.0 kΩ 1/10W 1% 0603 SMD
3	100 kΩ	R2, R3, R4	RES 100 kΩ 1/10W 1% 0603 SMD
4	0.022 µF	C25, C26, C27, C28	CAP CER 0.022 µF 50V X8R 10% 0603
2	0.047 µF	C23, C24	CAP CER 47000 pF 50V X7R 10% 0603
6	0.1 µF	C34,C35, C36, C37, C38, C39	CAP CER 0.1 µF 6.3V X5R 10% 0402
4	0.1 µF	C4, C5, C9, C10	CAP CER 0.1 µF 25V X7R 0603
1	0.22 µF	C11	CAP CER 0.22 µF 16V X7R 10% 0603
3	0.47 µF	C5, C7, C8	CAP CER 0.47 µF 10V X5R 10% 0603
4	10 µF	C30, C31, C32, C33	CAP CERAMIC 10 µF 6.3V X5R 0603
2	10 µF	C1, C2	CAP CERAMIC 10 µF 10V X5R 0805
1	10 µF	C3	CAP CER 10 µF 16V X5R 20% 1206
2	22 µF	C21, C22	CAP CER 22 µF 6.3V X5R 20% 0805
9	47 µF	C12, C13, C14, C15, C16, C17, C18, C19, C20	CAP CER 47 µF 10V X5R 1210
7	no value - not installed	C40, C41, C42, C43, C44, C45, C46	CAP 0603
1		U1	Audio Codec
1		U2	Single 2-Input Positive-AND Gate
1		U3	Dual 2-Input Positive-NAND Gate
1		U4	Dual-Output Low-Dropout (LDO) Voltage Regulators
1		U5	IC SERIAL EEPROM 64K 1.7V 8SOIC
1	600	L1	FERRITE CHIP 600 OHM 500MA 0805
1		LED1	LED THIN 635NM RED DIFF 0805 SMD
1		MK1	MIC CONDENSER ELECT OMNI -44DB or alternate
2		SW1, SW2	SWITCH SLIDE SPDT 30V.2A PC MNT
2		J7, J15	CONN JACK STEREO 5POS 3.5MM SMD
1		J14	CONN AUDIO JACK 3.5MM 4COND SMD
1		J12	Screw Terminal Block, 2 Position
4		J6, J8, J9, J13	Screw Terminal Block, 3 Position
2		J10, J11	CONN HEADER 3POS .100 VERT TIN

Qty	Value	Ref Des	Description
1		P3	10 Pin SMT Plug Header
1		J3	10 pin SMT Socket Header
2		P4, P5	20 Pin SMT Plug Header
4		J1, J2, J4, J5	20 pin SMT Socket Header
6	not installed	TP7, TP8, TP9, TP10, TP11, TP12	TEST POINT PC MINI .040"D RED
24	not installed	TP13, TP14, TP15, TP16, TP17, TP18, TP19, TP20, TP21, TP22, TP23, TP24, TP25, TP26, TP27, TP28, TP29, TP30, TP31, TP32, TP33, TP34, TP35, TP36	TEST POINT PC MINI .040"D WHITE
6		TP1, TP2, TP3, TP4, TP5, TP6	TEST POINT PC MULTI PURPOSE BLK
6		W10, W11, W12, W13, W14, W15	Bus Wire (18-22 Gauge)
12		W2, W3, W4, W5, W6, W16, W17, W18, W19, W20, W21, W22	2 Pin Thru-hole Plug Header (Jumper), 0 .1" spacing
4		W1, W7, W8, W9	3 Position Jumper , 0 .1" spacing

Appendix A USB-MODEVM Schematic

The schematic diagram for USB-MODEVM Interface Board is provided as a reference.

REVISION HISTORY		
REV	ENGINEERING CHANGE NUMBER	APPROVED



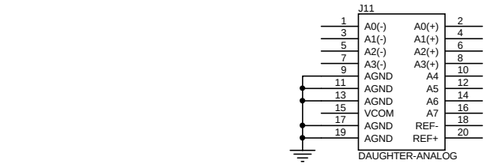
TEXAS INSTRUMENTS

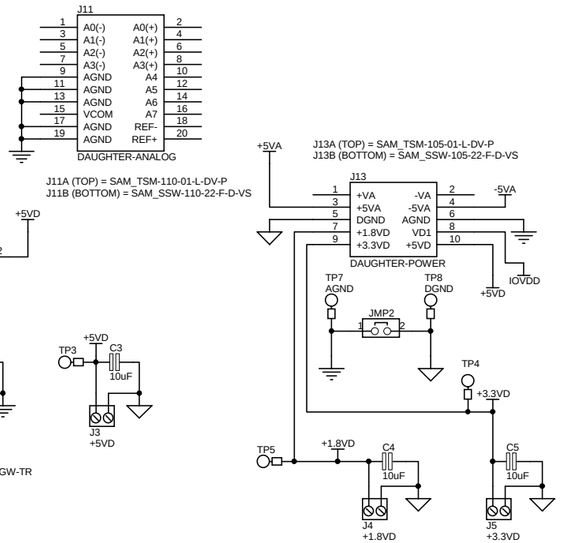
DATA ACQUISITION PRODUCTS
HIGH-PERFORMANCE ANALOG DIVISION
SEMICONDUCTOR GROUP

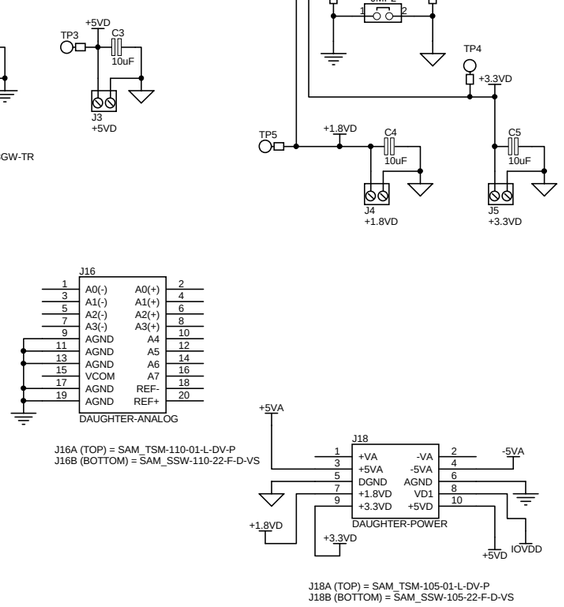
6730 SOUTH TUCSON BLVD., TUCSON, AZ 85706 USA

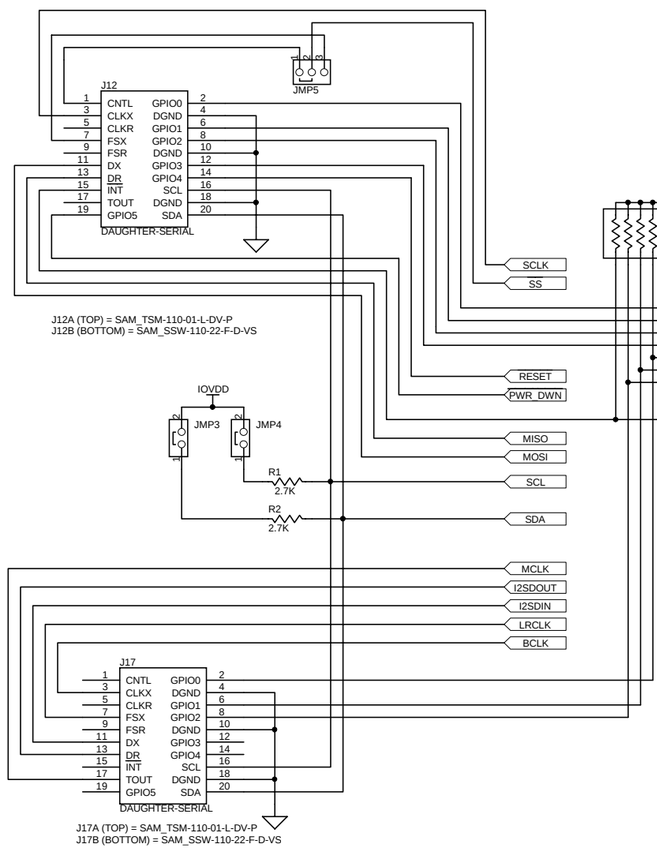
ENGINEER **FRYE D. ZERKETTS**	TITLE **untitled**		
DRAWN BY **I. C. SPOTTS**			
DOCUMENT CONTROL NO.**1234567**	SIZE **B**	DATE **dd MMM yyyy**	REV **A**
SHEET **1** OF **1**	FILE **???**		

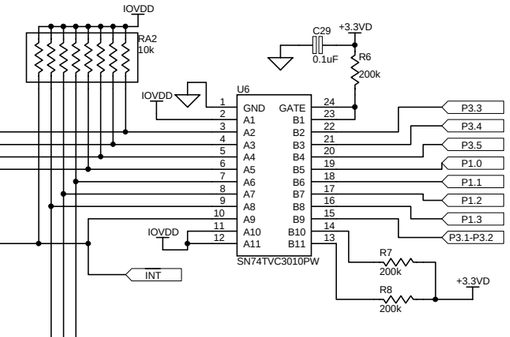
REVISION HISTORY		
REV	ENGINEERING CHANGE NUMBER	APPROVED





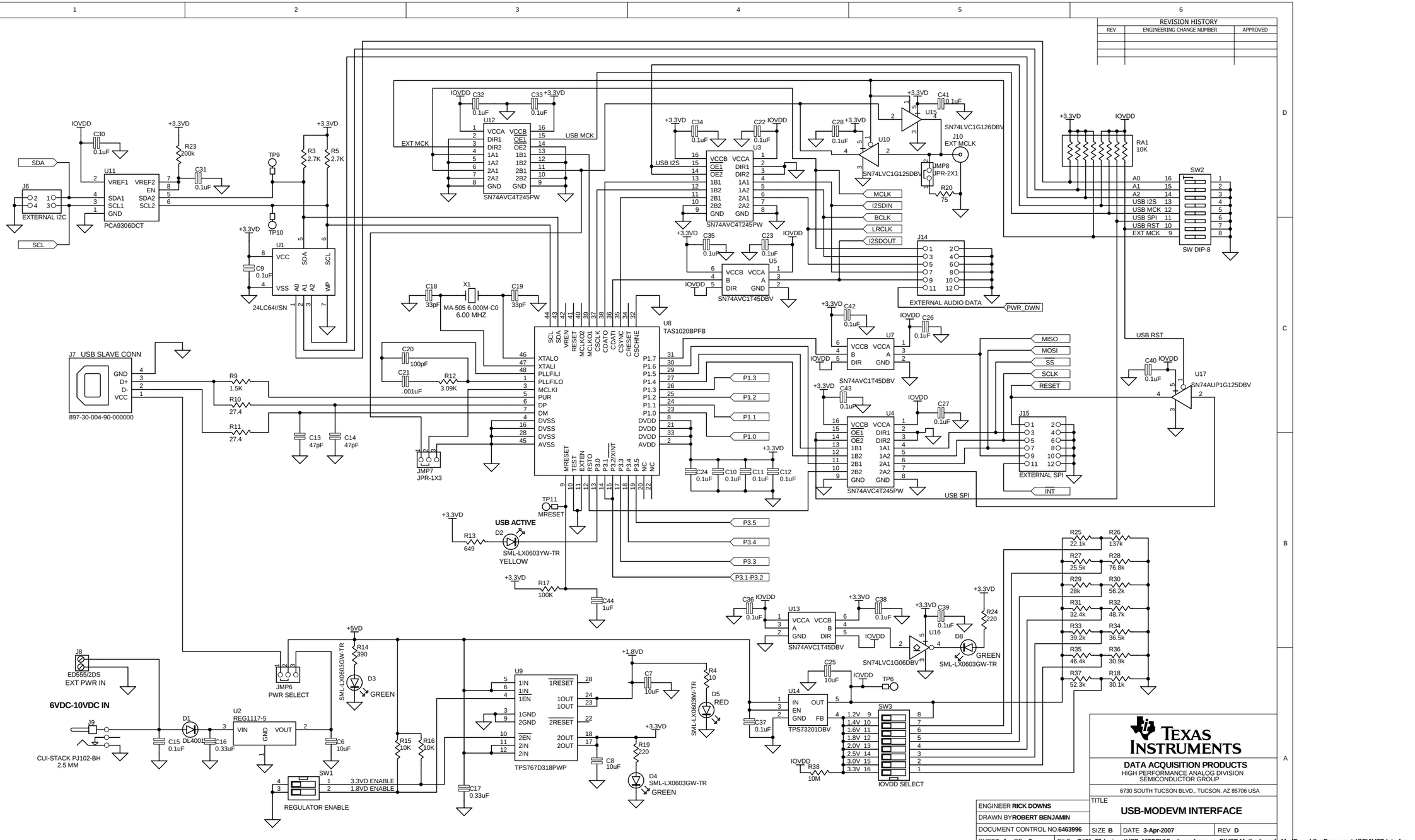






TEXAS INSTRUMENTS	
DATA ACQUISITION PRODUCTS HIGH-PERFORMANCE ANALOG DIVISION SEMICONDUCTOR GROUP	
6730 SOUTH TUCSON BLVD., TUCSON, AZ 85706 USA	
ENGINEER **RICK DOWNS**	TITLE **USB-MODEVM INTERFACE**
DRAWN BY **ROBERT BENJAMIN**	
DOCUMENT CONTROL NO. **6463996**	SIZE **B** DATE **3-Apr-2007** REV **D**
SHEET **2** OF **2**	FILE **C:\01_TI\designs\USB_MODEVM\usb-modevm_revD\USB Motherboard - ModEvm.ddb - Documents\SCH\Daughtercard Interface**

REVISION HISTORY		
REV	ENGINEERING CHANGE NUMBER	APPROVED



ENGINEER **RICK DOWNS**	
DRAWN BY **ROBERT BENJAMIN**	
DOCUMENT CONTROL NO. **6463996**	SIZE **B** DATE **3-Apr-2007** REV **D**
SHEET **1** OF **2**	FILE **C:\01_TI\designs\USB_MODEVM\usb-modevm_revD\USB Motherboard - ModEvm.ddb - Documents\SCH\USB Interface**

TEXAS INSTRUMENTS

DATA ACQUISITION PRODUCTS
HIGH PERFORMANCE ANALOG DIVISION
SEMICONDUCTOR GROUP

6730 SOUTH TUCSON BLVD., TUCSON, AZ 85706 USA

TITLE
USB-MODEVM INTERFACE

Appendix B USB-MODEVM Bill of Materials

The complete bill of materials for USB-MODEVM Interface Board is provided as a reference.

Table B-1. USB-MODEVM Bill of Materials

Designators	Description	Manufacturer	Mfg. Part Number
R4	10Ω 1/10W 5% Chip Resistor	Panasonic	ERJ-3GEYJ1300V
R10, R11	27.4Ω 1/16W 1% Chip Resistor	Panasonic	ERJ-3EKF27R4V
R20	75Ω 1/4W 1% Chip Resistor	Panasonic	ERJ-14NF75R0U
R19	220Ω 1/10W 5% Chip Resistor	Panasonic	ERJ-3GEYJ221V
R14, R21, R22	390Ω 1/10W 5% Chip Resistor	Panasonic	ERJ-3GEYJ391V
R13	649Ω 1/16W 1% Chip Resistor	Panasonic	ERJ-3EKF6490V
R9	1.5KΩ 1/10W 5% Chip Resistor	Panasonic	ERJ-3GEYJ1352V
R1–R3, R5–R8	2.7KΩ 1/10W 5% Chip Resistor	Panasonic	ERJ-3GEYJ272V
R12	3.09KΩ 1/16W 1% Chip Resistor	Panasonic	ERJ-3EKF3091V
R15, R16	10KΩ 1/10W 5% Chip Resistor	Panasonic	ERJ-3GEYJ1303V
R17, R18	100kΩ 1/10W 5%Chip Resistor	Panasonic	ERJ-3GEYJ1304V
RA1	10KΩ 1/8W Octal Isolated Resistor Array	CTS Corporation	742C163103JTR
C18, C19	33pF 50V Ceramic Chip Capacitor, ±5%, NPO	TDK	C1608C0G1H330J
C13, C14	47pF 50V Ceramic Chip Capacitor, ±5%, NPO	TDK	C1608C0G1H470J
C20	100pF 50V Ceramic Chip Capacitor, ±5%, NPO	TDK	C1608C0G1H101J
C21	1000pF 50V Ceramic Chip Capacitor, ±5%, NPO	TDK	C1608C0G1H102J
C15	0.1µF 16V Ceramic Chip Capacitor, ±10%, X7R	TDK	C1608X7R1C104K
C16, C17	0.33µF 16V Ceramic Chip Capacitor, ±20%, Y5V	TDK	C1608X5R1C334K
C9–C12, C22–C28	1µF 6.3V Ceramic Chip Capacitor, ±10%, X5R	TDK	C1608X5R0J1305K
C1–C8	10µF 6.3V Ceramic Chip Capacitor, ±10%, X5R	TDK	C3216X5R0J1306K
D1	50V, 1A, Diode MELF SMD	Micro Commercial Components	DL4001
D2	Yellow Light Emitting Diode	Lumex	SML-LX0603YW-TR
D3– D7	Green Light Emitting Diode	Lumex	SML-LX0603GW-TR
D5	Red Light Emitting Diode	Lumex	SML-LX0603IW-TR
Q1, Q2	N-Channel MOSFET	Zetex	ZXMN6A07F
X1	6MHz Crystal SMD	Epson	MA-505 6.000M-C0
U8	USB Streaming Controller	Texas Instruments	TAS1020BPFB
U2	5V LDO Regulator	Texas Instruments	REG1117-5
U9	3.3V/1.8V Dual Output LDO Regulator	Texas Instruments	TPS767D318PWP
U3, U4	Quad, 3-State Buffers	Texas Instruments	SN74LVC125APW
U5–U7	Single IC Buffer Driver with Open Drain o/p	Texas Instruments	SN74LVC1G07DBVR
U10	Single 3-State Buffer	Texas Instruments	SN74LVC1G125DBVR
U1	64K 2-Wire Serial EEPROM I^2C	Microchip	24LC64I/SN
	USB-MODEVM PCB	Texas Instruments	6463995
TP1–TP6, TP9–TP11	Miniature test point terminal	Keystone Electronics	5000
TP7, TP8	Multipurpose test point terminal	Keystone Electronics	5011
J7	USB Type B Slave Connector Thru-Hole	Mill-Max	897-30-004-90-000000
J13, J2–J5, J8	2-position terminal block	On Shore Technology	ED555/2DS
J9	2.5mm power connector	CUI Stack	PJ-102B
J130	BNC connector, female, PC mount	AMP/Tyco	414305-1
J131A, J132A, J21A, J22A	20-pin SMT plug	Samtec	TSM-110-01-L-DV-P
J131B, J132B, J21B, J22B	20-pin SMT socket	Samtec	SSW-110-22-F-D-VS-K
J133A, J23A	10-pin SMT plug	Samtec	TSM-105-01-L-DV-P
J133B, J23B	10-pin SMT socket	Samtec	SSW-105-22-F-D-VS-K
J6	4-pin double row header (2x2) 0.1"	Samtec	TSW-102-07-L-D
J134, J135	12-pin double row header (2x6) 0.1"	Samtec	TSW-106-07-L-D
JMP1–JMP4	2-position jumper, 0.1" spacing	Samtec	TSW-102-07-L-S

Table B-1. USB-MODEVM Bill of Materials (continued)

Designators	Description	Manufacturer	Mfg. Part Number
JMP8–JMP14	2-position jumper, 0.1" spacing	Samtec	TSW-102-07-L-S
JMP5, JMP6	3-position jumper, 0.1" spacing	Samtec	TSW-103-07-L-S
JMP7	3-position dual row jumper, 0.1" spacing	Samtec	TSW-103-07-L-D
SW1	SMT, half-pitch 2-position switch	C&K Division, ITT	TDA02H0SK1
SW2	SMT, half-pitch 8-position switch	C&K Division, ITT	TDA08H0SK1
	Jumper plug	Samtec	SNT-100-BK-T

Appendix C USB-MODEVM Protocol

C.1 USB-MODEVM Protocol

The USB-MODEVM is defined to be a Vendor-Specific class and is identified on the PC system as an NI-VISA device. Because the TAS1020B has several routines in its ROM which are designed for use with HID-class devices, HID-like structures are used, even though the USB-MODEVM is not an HID-class device. Data is passed from the PC to the TAS1020B using the control endpoint.

Data is sent in a HIDSETREPORT (see Table C-1).

Table C-1. USB Control Endpoint HIDSETREPORT Request

Part	Value	Description
bmRequestType	0x21	00100001
bRequest	0x09	SET_REPORT
wValue	0x00	don't care
wIndex	0x03	HID interface is index 3
wLength	calculated by host	
Data		Data packet as described in Table C-2.

The data packet consists of the following bytes, shown in Table C-2:

Table C-2. Data Packet Configuration

BYTE NUMBER	TYPE	DESCRIPTION
0	Interface	Specifies serial interface and operation. The two values are logically ORed. Operation: READ 0x00 WRITE 0x10 Interface: GPIO 0x08 SPI_16 0x04 I2C_FAST 0x02 I2C_STD 0x01 SPI_8 0x00
1	I^2C Slave Address	Slave address of I^2C device or MSB of 16-bit reg addr for SPI
2	Length	Length of data to write/read (number of bytes)
3	Register address	Address of register for I^2C or 8-bit SPI; LSB of 16-bit address for SPI
4..64	Data	Up to 60 data bytes could be written at a time. EP0 maximum length is 64. The return packet is limited to 42 bytes, so advise only sending 32 bytes at any one time.

Example usage:
Write two bytes (AA, 55) to device starting at register 5 of an I^2C device with address A0:

```
[0]    0x11
[1]    0xA0
[2]    0x02
[3]    0x05
[4]    0xAA
[5]    0x55
```

Do the same with a fast mode I²C device:

```
[0]     0x12
[1]     0xA0
[2]     0x02
[3]     0x05
[4]     0xAA
[5]     0x55
```

Now with an SPI device which uses an 8-bit register address:

```
[0]     0x10
[1]     0xA0
[2]     0x02
[3]     0x05
[4]     0xAA
[5]     0x55
```

Now, do a 16-bit register address, as found on parts like the TSC2101. Assume the register address (command word) is **0x10E0**:

```
[0]     0x14
[1]     0x10 → Note: the I²C address now serves as MSB of reg addr.
[2]     0x02
[3]     0xE0
[4]     0xAA
[5]     0x55
```

In each case, the TAS1020 returns, in an HID interrupt packet, the following:

[0] interface byte | status

status:

`REQ_ERROR 0x80`

`INTF_ERROR 0x40`

`REQ_DONE 0x20`

[1] for I²C interfaces, the I²C address as sent

for SPI interfaces, the read back data from SPI line for transmission of the corresponding byte

[2] length as sent

[3] for I²C interfaces, the reg address as sent

for SPI interfaces, the read back data from SPI line for transmission of the corresponding byte

[4..60] echo of data packet sent

If the command is sent with no problem, the returning byte [0] is the same as the sent one logically ORed with 0x20 - in the preceding first example, the returning packet is:

```
[0]    0x31
[1]    0xA0
[2]    0x02
[3]    0x05
[4]    0xAA
[5]    0x55
```

If for some reason the interface fails (for example, the I^2C device does not acknowledge), it comes back as:

```
[0]    0x51 → interface | INTF_ERROR
[1]    0xA0
[2]    0x02
[3]    0x05
[4]    0xAA
[5]    0x55
```

If the request is malformed, that is, the interface byte (byte [0]) takes on a value which is not as previously described, the return packet is:

```
[0]    0x93 → the user sent 0x13, which is not valid, so 0x93 returned
[1]    0xA0
[2]    0x02
[3]    0x05
[4]    0xAA
[5]    0x55
```

The preceding examples used writes. Reading is similar:
Read two bytes from device starting at register 5 of an I^2C device with address A0:

```
[0]    0x01
[1]    0xA0
[2]    0x02
[3]    0x05
```

The return packet is:

```
[0]    0x21
[1]    0xA0
[2]    0x02
[3]    0x05
[4]    0xAA
[5]    0x55
```

assuming that the values written starting at Register 5 were actually written to the device.

C.2 GPIO Capability

The USB-MODEVM has seven GPIO lines. Access them by specifying the interface to be 0x08, and then using the standard format for packets—but addresses are unnecessary. The GPIO lines are mapped into one byte (see Table C-3):

Table C-3. GPIO Pin Assignments

Bit 7	6	5	4	3	2	1	0
x	P3.5	P3.4	P3.3	P1.3	P1.2	P1.1	P1.0

Example: write P3.5 to a 1, set all others to 0:

```
[0]    0x18 → write, GPIO
[1]    0x00 → this value is ignored
[2]    0x01 → length - ALWAYS a 1
[3]    0x00 → this value is ignored
[4]    0x40 → 01000000
```

The user can also read back from the GPIO to see the state of the pins. Assume the previous example was just written to the port pins.

Example: read the GPIO

```
[0]    0x08 → read, GPIO
[1]    0x00 → this value is ignored
[2]    0x01 → length - ALWAYS a 1
[3]    0x00 → this value is ignored
```

The return packet is:

```
[0]    0x28
[1]    0x00
[2]    0x01
[3]    0x00
[4]    0x40
```

C.3 Writing Scripts

A script is simply a text file that contains data to send to the serial control buses.

Each line in a script file is one command. No provision is made for extending lines beyond one line, except for the **>** command. A line is terminated by a carriage return.

The first character of a line is the command. Commands are:

i Set interface bus to use
r Read from the serial control bus
w Write to the serial control bus
> Extend repeated write commands to lines below a **w**
Comment
b Break
d Delay
f Wait for Flag

The first command, **i**, sets the interface to use for the commands to follow. This command must be followed by one of the following parameters:

i2cstd	Standard mode I^2C bus
i2cfast	Fast mode I^2C bus
spi8	SPI bus with 8-bit register addressing
spi16	SPI bus with 16-bit register addressing
gpio	Use the USB-MODEVM GPIO capability

For example, if a fast mode I^2C bus is to be used, the script begins with:

i i2cfast

A double quoted string of characters following the **b** command can be added to provide information to the user about each breakpoint. When the script is executed, the software's command handler halts as soon as a breakpoint is detected and displays the string of characters within the double quotes.

The Wait for Flag command, **f,** reads a specified register and verifies if the bitmap provided with the command matches the data being read. If the data does not match, the command handler retries for up to 200 times. This feature is useful when switching buffers in parts that support the adaptive filtering mode. The command f syntax follows:

```
f [i2c address] [register] [D7][D6][D5][D4][D3][D2][D1][D0]

where 'i2c address' and 'register' are in hexadecimal format
and 'D7' through 'D0' are in binary format with values of 0,
1 or X for don't care.
```

Anything following a comment command # is ignored by the parser, provided that it is on the same line.

The delay command **d** allows the user to specify a time, in milliseconds, that the script pauses before proceeding. **The delay time is entered in decimal format.**

A series of byte values follows either a read or write command. Each byte value is expressed in hexadecimal, and each byte must be separated by a space. Commands are interpreted and sent to the TAS1020B by the program using the protocol described in Section C.1.

The first byte following an **r** (read) or **w** (write) command is the I^2C slave address of the device (if I^2C is used) or the first data byte to write (if SPI is used—note that SPI interfaces are not standardized on protocols, so the meaning of this byte varies with the device being addressed on the SPI bus). The second byte is the starting register address that data will be written to (again, with I^2C; SPI varies—see Section C.1 for additional information about what variations may be necessary for a particular SPI mode). Following these two bytes are data, if writing; if reading, the third byte value is the number of bytes to read, (expressed in hexadecimal).

For example, to write the values 0xAA 0x55 to an I^2C device with a slave address of 0x30, starting at a register address of 0x03, the user writes:

```
#example script

i i2cfast

w 30 03 AA 55

r 30 03 02
```

This script begins with a comment, specifies that a fast I^2C bus will be used, then writes *0xAA 0x55* to the I^2C slave device at address 0x30, writing the values into registers 0x03 and 0x04. The script then reads back two bytes from the same device starting at register address 0x03. Note that the slave device value does not change. It is unnecessary to set the R/$\overline{W}$ bit for I^2C devices in the script; the read or write commands does that.

If extensive repeated write commands are sent and commenting is desired for a group of bytes, the **>** command can be used to extend the bytes to other lines that follow. A usage example for the **>** command follows:

```
#example script for '>' command

i i2cfast

# Write AA and BB to registers 3 and 4, respectively

w 30 03 AA BB

# Write CC, DD, EE and FF to registers 5, 6, 7 and 8, respectively

> CC DD EE FF

# Place a commented breakpoint

b "AA BB CC DD EE FF was written, starting at register 3"

# Read back all six registers, starting at register 3

r 30 03 06
```

The following example demonstrates usage of the Wait for Flag command, **f**:

```
#example script for 'wait for flag' command

i i2cfast

# Switch to Page 44

w 30 00 2C

# Switch buffers

w 30 01 05

# Wait for bit D0 to clear. 'x' denotes a don't care.

f 30 01 xxxxxxx0
```

Any text editor can be used to write these scripts; Jedit is an editor that is highly recommended for general usage. For more information, go to: http://www.jedit.org.

Once the script is written, it can be used in the command window by running the program, and then selecting *Open Script File...* from the File menu. Locate the script and open it. The script is then displayed in the command buffer. The user can also edit the script once it is in the buffer and save it by selecting *Save Script File...* from the File menu.

Once the script is in the command buffer, it can be executed by pressing the *Execute Command Buffer* button. If there are breakpoints in the script, the script executes to that point, and the user is presented with a dialog box with a button to press to continue executing the script. When ready to proceed, push that button and the script continues.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DLP® Products	www.dlp.com	Broadband	www.ti.com/broadband
DSP	dsp.ti.com	Digital Control	www.ti.com/digitalcontrol
Clocks and Timers	www.ti.com/clocks	Medical	www.ti.com/medical
Interface	interface.ti.com	Military	www.ti.com/military
Logic	logic.ti.com	Optical Networking	www.ti.com/opticalnetwork
Power Mgmt	power.ti.com	Security	www.ti.com/security
Microcontrollers	microcontroller.ti.com	Telephony	www.ti.com/telephony
RFID	www.ti-rfid.com	Video & Imaging	www.ti.com/video
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265