

TLV320AIC32EVM and TLV320AIC32EVM-PDK User's Guide

This user's guide describes the characteristics, operation, and use of the TLV320AIC32EVM, both by itself and as part of the TLV320AIC32EVM-PDK. This evaluation module (EVM) is a complete stereo audio codec with several inputs and outputs, extensive audio routing, mixing and effects capabilities. A complete circuit description, schematic diagram and bill of materials are also included.

The following related documents are available through the Texas Instruments web site at www.ti.com.

EVM-Compatible Device Data Sheets

Device	Literature Number
TLV320AIC32	SLAS479
TAS1020B	SLES025
REG1117-3.3	SBVS001
TPS767D318	SLVS209
SN74LVC125A	SCAS290
SN74LVC1G125	SCES223
SN74LVC1G07	SCES296

Contents

1	EVM Overview	2
2	Analog Interface.....	3
3	Digital Interface	4
4	Power Supplies	5
5	EVM Operation.....	6
6	Kit Operation	7
7	EVM Bill of Materials.....	36
Appendix A	TLV320AIC32EVM Schematic	40
Appendix B	USB-MODEVM Schematic	41

List of Figures

1	TLV320AIC32EVM-PDK Block Diagram	8
2	Default Software Screen	10
3	Audio Generator Screen	12
4	Audio Analyzer Screen	13
5	Audio Input Tab	14
6	Audio Interface Tab	16
7	Clocks Tab	17

I²S, I²C are trademarks of Koninklijke Philips Electronics N.V.
Windows is a trademark of Microsoft Corporation.
SPI is a trademark of Motorola, Inc.
LabView is a trademark of National Instruments.
All trademarks are the property of their respective owners.

8	AGC Tab	19
9	Filters Tab	20
10	ADC Highpass Filter Settings	21
11	Enabling Filters	21
12	Shelf Filters	22
13	EQ Filters	22
14	Analog Simulation Filters	23
15	Preset Filters	23
16	De-emphasis Filters	24
17	User Filters	25
18	3D Effect Settings	25
19	DAC/Line Outputs Tab	26
20	Output Stage Configuration Tab	28
21	High Power Outputs Tab	29
22	Command Line Interface Tab	30
23	File Menu	31

List of Tables

1	Analog Interface Pinout	3
2	Alternate Analog Connectors.....	4
3	Digital Interface Pinout	4
4	Power Supply Pin Out.....	5
5	List of Jumpers.....	7
6	USB-MODEVM SW2 Settings	9
7	USB Control Endpoint HIDSETREPORT Request	31
8	Data Packet Configuration	32
9	GPIO Pin Assignments	34
10	TLV320AIC32EVM Bill of Materials.....	37
11	USB-MODEVM Bill of Materials	38

1 EVM Overview

1.1 Features

- Full-featured evaluation board for the TLV320AIC32 stereo audio codec.
- Modular design for use with a variety of digital signal processor (DSP) and microcontroller interface boards.

The TLV320AIC32EVM-PDK is a complete evaluation kit, which includes a universal serial bus (USB)-based motherboard and evaluation software for use with a personal computer running Microsoft Windows™ operating systems (Win2000 or XP).

1.2 Introduction

The TLV320AIC32EVM is in Texas Instruments' modular EVM form factor, which allows direct evaluation of the device performance and operating characteristics, and eases software development and system prototyping. This EVM is compatible with the 5-6K Interface Evaluation Module ([SLAU104](#)) and the HPA-MCUINTERFACE ([SLAU106](#)) from Texas Instruments and additional third-party boards that support the TI Modular EVM format.

The TLV320AIC32EVM-PDK is a complete evaluation/demonstration kit, which includes a USB-based motherboard called the USB-MODEVM Interface board and evaluation software for use with a personal computer running Microsoft Windows operating systems.

2 Analog Interface

For maximum flexibility, the TLV320AIC32EVM is designed for easy interfacing to multiple analog sources. Samtec part numbers SSW-110-22-F-D-VS-K and TSM-110-01-T-DV-P provide a convenient 10-pin dual row header/socket combination at J1 and J2. These headers/sockets provide access to the analog input and output pins of the device. Consult Samtec at www.samtec.com or call 1-800-SAMTEC-9 for a variety of mating connector options. Table 1 summarizes the analog interface pinout for the TLV320AIC32EVM.

Table 1. Analog Interface Pinout

PIN NUMBER	SIGNAL	DESCRIPTION
J1.1	HPLCOM	High Power Output Driver (Left Minus or Multifunctional)
J1.2	HPLOUT	High Power Output Driver (Left Plus)
J1.3	HPRCOM	High Power Output Driver (Right Minus or Multifunctional)
J1.4	HPROUT	High Power Output Driver (Right Plus)
J1.5	LINE1L	Left Input 1
J1.6	LINE1R	Right Input 1
J1.7	LINE2L	Left Input 2
J1.8	LINE2R	Right Input 2
J1.9	AGND	Analog Ground
J1.10	MIC3L	Left Input 3
J1.11	AGND	Analog Ground
J1.12	MIC3R	Right Input 3
J1.13	AGND	Analog Ground
J1.14	MICBIAS	Microphone Bias Voltage Output
J1.15	NC	Not Connected
J1.16	NC	Not Connected
J1.17	AGND	Analog Ground
J1.18	NC	Not Connected
J1.19	AGND	Analog Ground
J1.20	NC	Not Connected
J2.1	NC	Not Connected
J2.2	NC	Not Connected
J2.3	NC	Not Connected
J2.4	NC	Not Connected
J2.5	NC	Not Connected
J2.6	NC	Not Connected
J2.7	LEFT_LOP	Left Line Output (Plus)
J2.8	LEFT_LOM	Left Line Output (Minus)
J2.9	AGND	Analog Ground
J2.10	RIGHT_LOP	Right Line Output (Plus)
J2.11	AGND	Analog Ground
J2.12	RIGHT_LOM	Right Line Output (Minus)
J2.13	AGND	Analog Ground
J2.14	NC	Not Connected
J2.15	NC	Not Connected
J2.16	NC	Not Connected
J2.17	AGND	Analog Ground
J2.18	NC	Not Connected
J2.19	AGND	Analog Ground
J2.20	NC	Not Connected

In addition to the analog headers, the analog inputs and outputs may also be accessed through alternate connectors, either screw terminals or audio jacks. The stereo microphone input is also tied to J8 and the stereo headphone output (the HP set of outputs) is available at J9.

[Table 2](#) summarizes the screw terminals available on the TLV320AIC32EVM.

Table 2. Alternate Analog Connectors

DESIGNATOR	PIN 1	PIN 2	PIN3
J6	AGND	LINE1R	LINE1L
J7	MIC3L	MIC3R	AGND
J10	(+) LEFT_LOP	(-) LEFT_LOM	
J11	(+) RIGHT_LOP	(-) RIGHT_LOM	
J12	(+) HPLOUT	(-) HPLCOM	AGND
J13	(+) HPROUT	(-) HPRCOM	AGND
J14	AGND	LINE2R	LINE2L

3 Digital Interface

The TLV320AIC32EVM is designed to easily interface with multiple control platforms. Samtec part numbers SSW-110-22-F-D-VS-K and TSM-110-01-T-DV-P provide a convenient 10-pin dual row header/socket combination at J4 and J5. These headers/sockets provide access to the digital control and serial data pins of the device. Consult Samtec at www.samtec.com or call 1-800- SAMTEC-9 for a variety of mating connector options. [Table 3](#) summarizes the digital interface pinout for the TLV320AIC32EVM.

Table 3. Digital Interface Pinout

PIN NUMBER	SIGNAL	DESCRIPTION
J4.1	NC	Not Connected
J4.2	NC	Not Connected
J4.3	NC	Not Connected
J4.4	DGND	Digital Ground
J4.5	NC	Not Connected
J4.6	NC	Not Connected
J4.7	NC	Not Connected
J4.8	RESET INPUT	Reset signal input to AIC32EVM
J4.9	NC	Not Connected
J4.10	DGND	Digital Ground
J4.11	NC	Not Connected
J4.12	NC	Not Connected
J4.13	NC	Not Connected
J4.14	AIC32 /RESET	Reset
J4.15	NC	Not Connected
J4.16	SCL	I ² C Serial Clock
J4.17	NC	Not Connected
J4.18	DGND	Digital Ground
J4.19	NC	Not Connected
J4.20	SDA	I ² C Serial Data Input/Output
J5.1	NC	Not Connected
J5.2	NC	Not Connected
J5.3	BCLK	Audio Serial Data Bus Bit Clock (Input/Output)
J5.4	DGND	Digital Ground

Table 3. Digital Interface Pinout (continued)

PIN NUMBER	SIGNAL	DESCRIPTION
J5.5	NC	Not Connected
J5.6	NC	Not Connected
J5.7	WCLK	Audio Serial Data Bus Word Clock (Input/Output)
J5.8	NC	Not Connected
J5.9	NC	Not Connected
J5.10	DGND	Digital Ground
J5.11	DIN	Audio Serial Data Bus Data Input (Input)
J5.12	NC	Not Connected
J5.13	DOUT	Audio Serial Data Bus Data Output (Output)
J5.14	NC	Not Connected
J5.15	NC	Not Connected
J5.16	SCL	I ² C Serial Clock
J5.17	MCLK	Master Clock Input
J5.18	DGND	Digital Ground
J5.19	NC	Not Connected
J5.20	SDA	I ² C Serial Data Input/Output

Note that J5 comprises the signals needed for an I²S™ serial digital audio interface; the control interface (I²C™ and RESET) signals are routed to J4. I²C is actually routed to both connectors; however, the device is connected only to J4.

4 Power Supplies

J3 provides connection to the common power bus for the TLV320AIC32EVM. Power is supplied on the pins listed in [Table 4](#).

Table 4. Power Supply Pin Out

SIGNAL	PIN NUMBER		SIGNAL
NC	J3.1	J3.2	NC
+5VA	J3.3	J3.4	NC
DGND	J3.5	J3.6	AGND
DVDD (1.8V)	J3.7	J3.8	NC
IOVDD (3.3V)	J3.9	J3.10	NC

The TLV320AIC32EVM-PDK motherboard (the USB-MODEVM Interface board) supplies power to J3 of the TLV320AIC32EVM. Power for the motherboard is supplied either through its USB connection or via terminal blocks on that board.

4.1 Stand-Alone Operation

When used as a stand-alone EVM, power can be applied to J3 directly. The user must be sure to reference the supplies to the appropriate grounds on that connector.

CAUTION

Verify that all power supplies are within the safe operating limits shown on the [TLV320AIC32 data sheet](#) before applying power to the EVM.

4.2 USB-MODEVM Interface Power

The USB-MODEVM Interface board can be powered from several different sources:

- USB
- 6VDC-10VDC AC/DC external wall supply (not included)
- Lab power supply

When powered from the USB connection, JMP6 should have a shunt from pins 1–2 (this is the default factory configuration). When powered from 6V-10VDC, either through the J8 terminal block or the J9 barrel jack, JMP6 should have a shunt installed on pins 2-3. If power is applied in any of these ways, onboard regulators generate the required supply voltages and no further power supplies are necessary.

If lab supplies are used to provide the individual voltages required by the USB-MODEVM Interface, JMP6 should have no shunt installed. Voltages are then applied to J2 (+5VA), J3 (+5VD), J4 (+1.8VD), and J5 (+3.3VD). The +1.8VD and +3.3VD can also be generated on the board by the onboard regulators from the +5VD supply; to enable this configuration, the switches on SW1 need to be set to enable the regulators by placing them in the ON position (lower position, looking at the board with text reading right-side up). If +1.8VD and +3.3VD are supplied externally, disable the onboard regulators by placing SW1 switches in the OFF position.

Each power supply voltage has an LED (D1-D7) that lights when the power supplies are active.

5 EVM Operation

This section provides information on the analog input and output, digital control, and general operating conditions for the TLV320AIC32EVM.

5.1 Analog Input

The analog input sources can be applied directly to J1 (top or bottom side) or through signal conditioning modules available for the modular EVM system.

The analog inputs may also be accessed through J8 and screw terminals J6, J7, and J14.

5.2 Analog Output

The analog outputs from the TLV320AIC32 are available on J1 and J2 (top or bottom). They also may be accessed through J9, J10, J11, J12, and J13.

5.3 Digital Control

The digital control signals can be applied directly to J4 and J5 (top or bottom side). The modular TLV320AIC32EVM can also be connected directly to a DSP interface board, such as the 5-6KINTERFACE or HPA-MCUINTERFACE, or to the USB-MODEVM Interface board if purchased as part of the TLV320AIC32EVM-PDK. See the product folder for this [EVM](#) or the [TLV320AIC32](#) for a current list of compatible interface and/or accessory boards.

5.4 Default Jumper Locations

Table 5 lists the jumpers found on the EVM and their respective factory default conditions.

Table 5. List of Jumpers

JUMPER	DEFAULT POSITION	JUMPER DESCRIPTION
JMP1	Installed	Connects analog and digital grounds
JMP2	Open	Selects on-board EEPROM as firmware source
JMP3	Installed	Connects on-board Mic to Left Microphone Input
JMP4	Installed	Connects on-board Mic to Right Microphone Input
JMP5	Installed	Provides a means of measuring IOVDD current
JMP6	Installed	Provides a means of measuring DVDD current
JMP7	Installed	Provides a means of measuring DRVDD current
JMP8	Installed	Provides a means of measuring AVDD_DAC current
JMP9	Open	When installed, allows the USB-MODEVM to hardware reset the device under user control
JMP10	2-3	When connecting 2-3, mic bias comes from the MICBIAS pin on the device; when connecting 1-2, mic bias is supplied from the power supply through a resistor, which the user must install.
JMP11	Installed	When installed, shorts across the output capacitor on HPLOUT; remove this jumper if using AC-coupled output drive
JMP12	Installed	When installed, shorts HPLCOM and HPRCOM. Use only if these signals are set to constant VCM.
JMP13	Installed	When installed, shorts across the output capacitor on HPLCOM; remove this jumper if using AC-coupled output drive
JMP14	Installed	When installed, shorts across the output capacitor on HPROUT; remove this jumper if using AC-coupled output drive
JMP15	Installed	When installed, shorts across the output capacitor on HPRCOM; remove this jumper if using AC-coupled output drive

6 Kit Operation

This section provides information on using the TLV320AIC32EVM-PDK, including set up, program installation, and program usage.

6.1 TLV320AIC32EVM-PDK Block Diagram

A block diagram of the TLV320AIC32EVM-PDK is shown in Figure 1. The evaluation kit consists of two circuit boards connected together. The motherboard is designated as the USB-MODEVM Interface board, while the daughtercard is the TLV320AIC32EVM described previously in this manual.

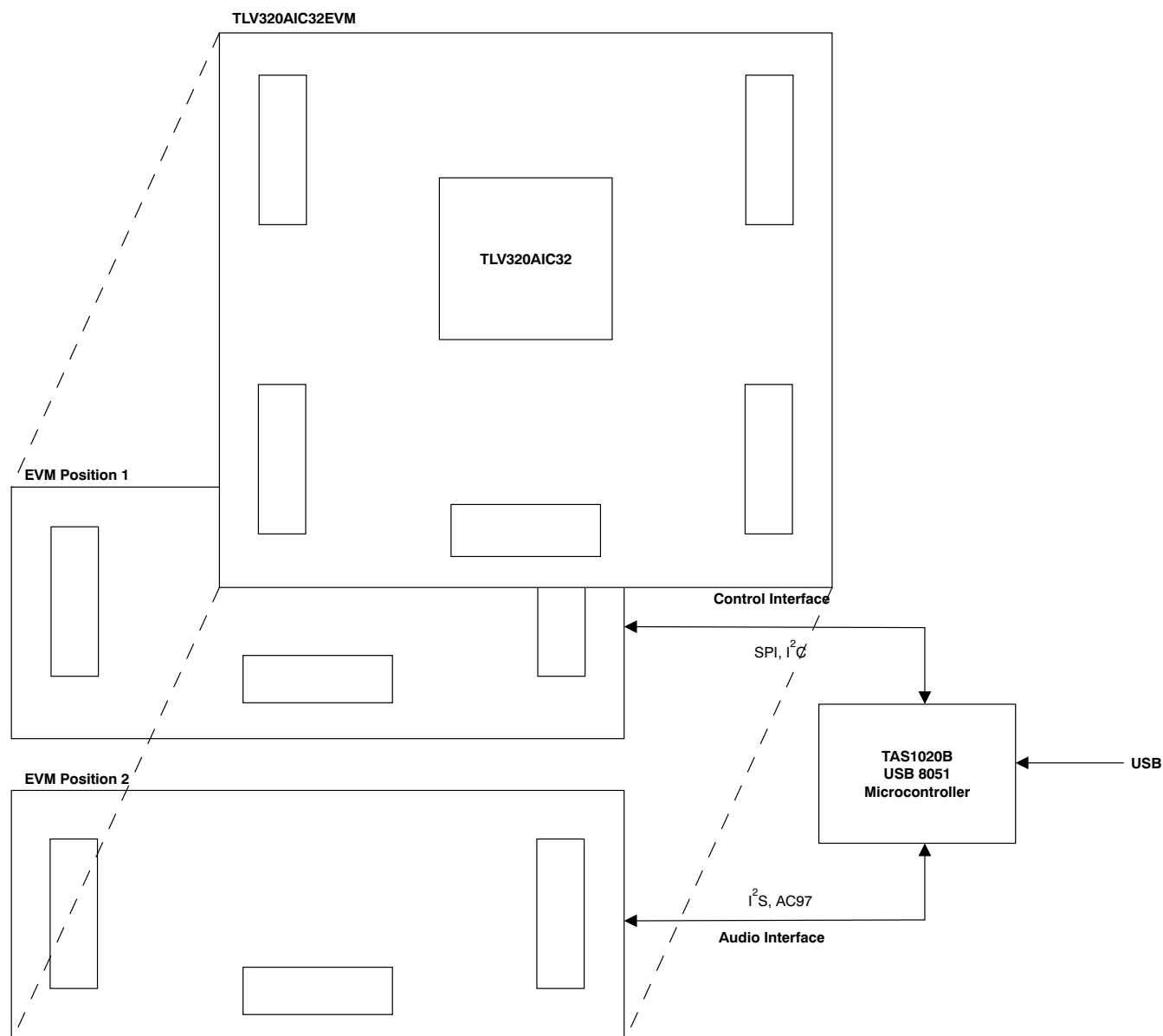


Figure 1. TLV320AIC32EVM-PDK Block Diagram

The USB-MODEVM Interface board is intended to be used in USB mode, where control of the installed EVM is accomplished using the onboard USB controller device. Provision is made, however, for driving all the data buses (I²C, SPI™, I²S/AC97) externally. The source of these signals is controlled by SW2 on the USB-MODEVM. Refer to Table 6 for details on the switch settings.

Table 6. USB-MODEVM SW2 Settings

SW-2 SWITCH NUMBER	LABEL	SWITCH DESCRIPTION
1	A0	USB-MODEVM EEPROM I ² C Address A0 ON: A0 = 0 OFF: A0 = 1
2	A1	USB-MODEVM EEPROM I ² C Address A1 ON: A1 = 0 OFF: A1 = 1
3	A2	USB-MODEVM EEPROM I ² C Address A2 ON: A2 = 0 OFF: A2 = 1
4	USB I ² S	I ² S Bus Source Selection ON: I ² S Bus connects to TAS1020 OFF: I ² S Bus connects to USB-MODEVM J14
5	USB MCK	I ² S Bus MCLK Source Selection ON: MCLK connects to TAS1020 OFF: MCLK connects to USB-MODEVM J14
6	USB SPI	SPI Bus Source Selection ON: SPI Bus connects to TAS1020 OFF: SPI Bus connects to USB-MODEVM J15
7	USB RST	RST Source Selection ON: EVM Reset Signal comes from TAS1020 OFF: EVM Reset Signal comes from USB-MODEVM J15
8	EXT MCK	External MCLK Selection ON: MCLK Signal is provided from USB-MODEVM J10 OFF: MCLK Signal comes from either selection of SW2-5

For use with the TLV320AIC32EVM, SW-2 positions 1 through 7 should be set to ON, while SW-2.8 should be set to OFF.

6.2 Installation

Ensure that the TLV320AIC32EVM is installed on the USB-MODEVM Interface board, aligning J1, J2, J3, J4, J5 with the corresponding connectors on the USB-MODEVM.

Place the CD-ROM into your PC CD-ROM drive. Locate the **Setup** program on the disk, and start it. The Setup program will install the TLV320AIC32 Evaluation software on your PC.

After the main program is installed, the NI-VISA Runtime installer will automatically run. This software allows the program to communicate with the USB-MODEVM.

When the installation completes, click *Finish* on the TLV320AIC32EVM installer window. You may be prompted to restart your computer.

When installation is complete, attach a USB cable from your PC to the USB-MODEVM Interface board. As configured at the factory, the board will be powered from the USB interface, so the power indicator LEDs on the USB-MODEVM should light. Once this connection is established, launch the TLV320AIC32 Evaluation software on your PC.

Kit Operation

The software should automatically find the TLV320AIC32EVM, and a screen similar to the one in [Figure 2](#) should appear.

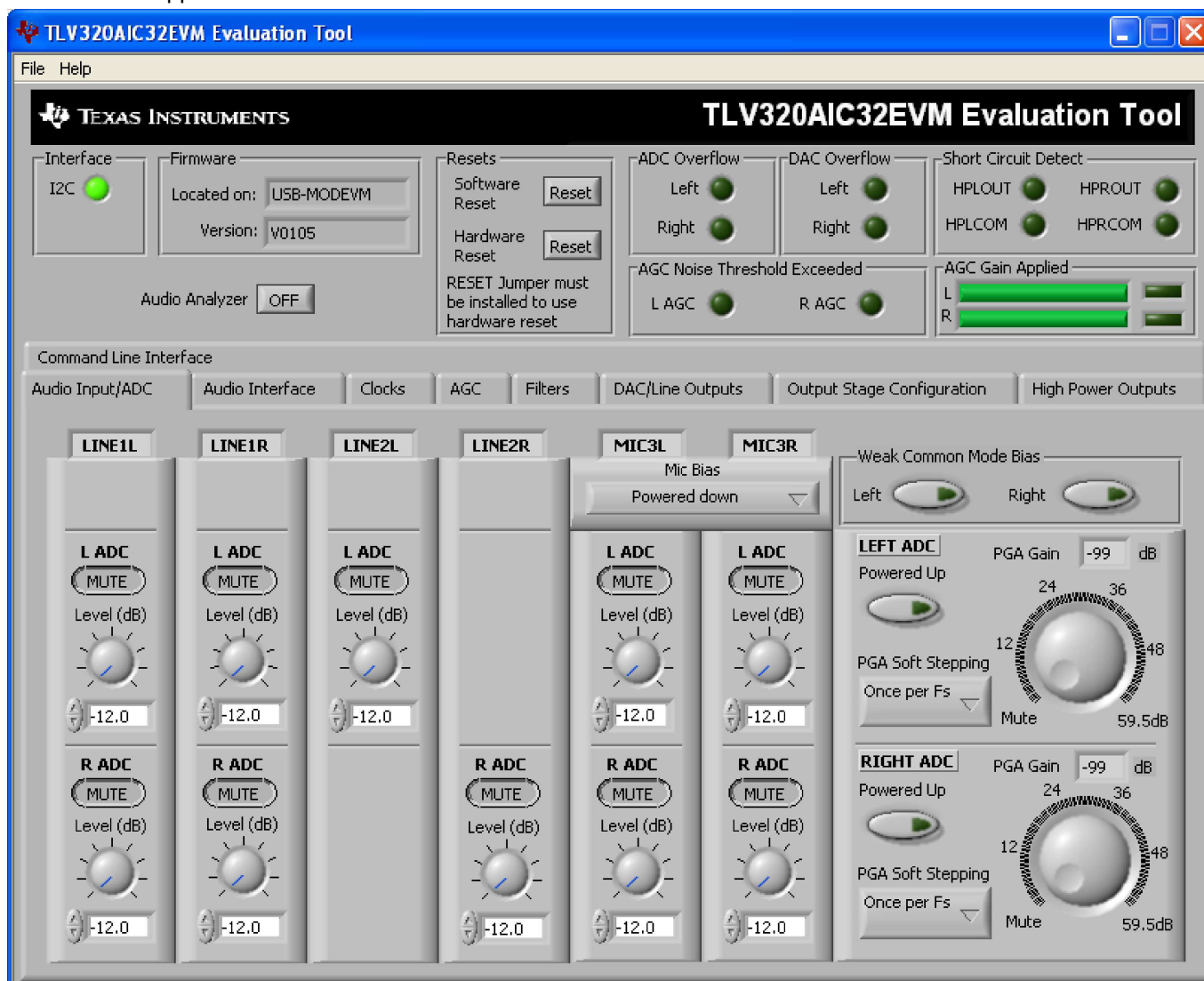


Figure 2. Default Software Screen

6.3 USB-MODEVM Interface Board

The simple diagram shown in [Figure 1](#) shows only the basic features of the USB-MODEVM Interface board. The board is built around a TAS1020B streaming audio USB controller with an 8051-based core. The board features two positions for modular EVMs, or one double-wide serial modular EVM may be installed.

Since the TLV320AIC32EVM is a double-wide modular EVM, it is installed with connections to both EVM positions, which connects the TLV320AIC32 digital control interface to the I²C port realized using the TAS1020B, as well as the TAS1020B digital audio interface.

In the factory configuration, the board is ready to use with the TLV320AIC32EVM. To view all the functions and configuration options available on the USB-MODEVM board, see the USB-MODEVM Interface Board schematic in [Appendix B](#).

6.4 Program Description

After the TLV320AIC32EVM-PDK software installation (described in [Section 6.2](#)) is complete, evaluation and development with the TLV320AIC32 can begin.

6.5 Indicators and Main Screen Controls

[Figure 2](#) illustrates the indicators and controls near the top of the software screen display, and a large tabbed interface below. This section discusses the controls above this tabbed section.

At the top left of the screen is an **Interface** indicator. The TLV320AIC32 has an I²C interface. The indicator is lit after the program begins.

To the right of the Interface indicator is a group box called **Firmware**. This box indicates where the firmware being used is operating from—in this release, the firmware is on the USB-MODEVM, so you should see *USB-MODEVM* in the box labeled **Located On:**. The version of the firmware appears in the **Version** box below this.

To the right, the next group box contains controls for resetting the TLV320AIC32. A software reset can be done by writing to a register in the TLV320AIC32; the writing is accomplished by pushing the button labeled **Software Reset**. The TLV320AIC32 also may be reset by toggling a pin on the TLV320AIC32, which is done by pushing the **Hardware Reset** button.

CAUTION

In order to perform a hardware reset, the RESET jumper (JMP9) must be installed and SW2-7 on the USB-MODEVM must be turned OFF. Failure to do either of these steps results in not generating a hardware reset or causing unstable operation of the EVM, which may require cycling power to the USB-MODEVM.

The **ADC Overflow** and **DAC Overflow** indicators light when the overflow flags are set in the TLV320AIC32. These indicators, as well as the other indicators on this panel, update only when the software's front panel is inactive, once every 20ms. Below these indicators are other indicators that show when the AGC noise threshold is exceeded. To the far right on this screen, the short-circuit indicators show when a short-circuit condition is detected, if this feature has been enabled. Below the short-circuit indicators is a bar graph that shows the amount of gain which has been applied by the AGC, and indicators that light when the AGC is saturated.

6.5.1 Audio Analyzer

Near the left side of the screen is a button labeled **Audio Analyzer**; this button can be set to ON or OFF. Pressing the button to turn it *ON* opens another window (see [Figure 3](#)). This feature provides the ability to generate signals to be sent to the TLV320AIC32 DACs, as well as viewing and analyzing signals read by the TLV320AIC32 ADCs. This ability to view and process the real-time streaming USB audio is a demanding task. Use of the Audio Analyzer feature requires a PC with at least 512MB of memory and reasonable processor speed (> 1GHz); computers with inadequate resources could still use the Audio Analyzer to generate signals for the DACs, but will be unable to process signals from the ADCs because the FFT, distortion analysis, and signal-to-noise ratio analysis will not be able to keep up with the data processing requirements.

The Audio Analyzer features two tabs. The front tab, shown in [Figure 3](#) and titled *Generator*, creates digital waveforms to send to the DACs. When first started, the function will be set to *SNR (Output Zeros)* which feeds only zero codes to the DAC. This function is commonly performed to test for the noise floor of the DAC.

The second function available is *THD (-1dB sinewave)*. This function sends a sinewave at a coherent frequency to the 44.1kHz sample rate to the DACs; this function is commonly used for testing THD+N.

These first two functions do not require any further settings; therefore, the frequency and amplitude knobs below the function selector are not used. Selecting a function of *Function Generator* allows the choice of waveform shape by using the pull-down menu next to the function selector, and the frequency and amplitude of that signal can be varied using the knobs below.

The output waveforms for both left and right channels are displayed in the graph at the bottom of the screen in [Figure 3](#).

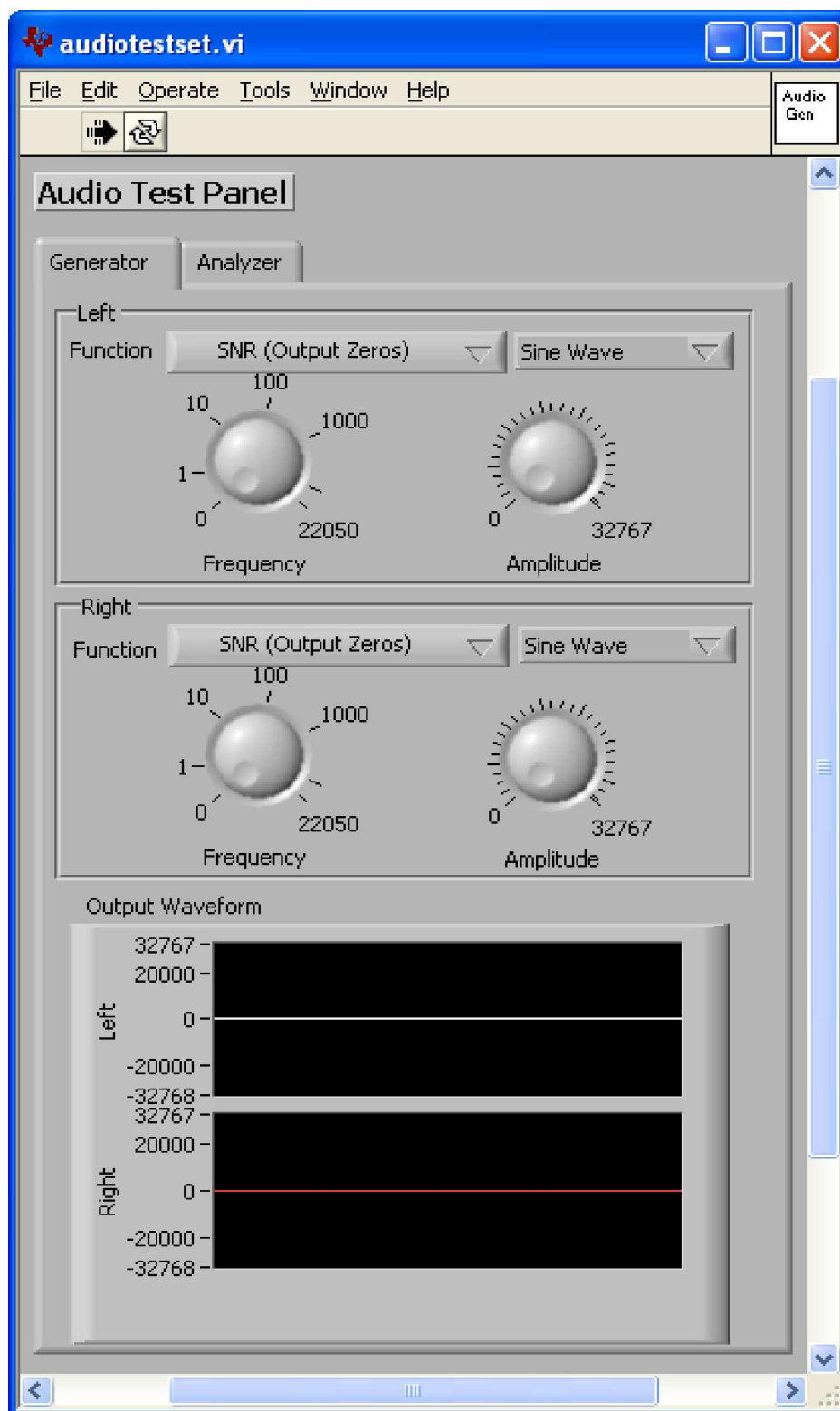


Figure 3. Audio Generator Screen

The second tab, titled **Analyzer** (Figure 4), handles the display and analysis of data from the ADCs.

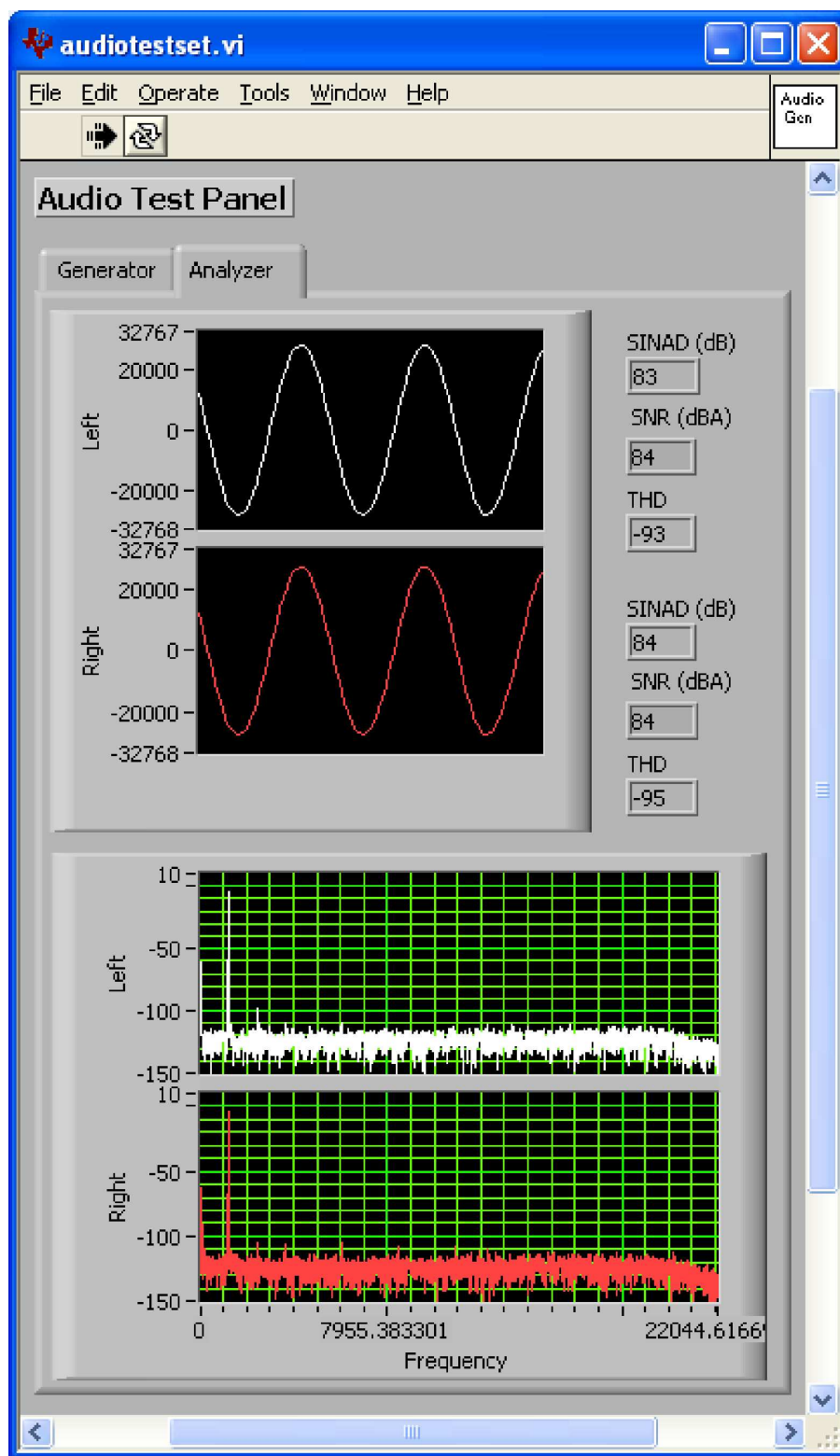


Figure 4. Audio Analyzer Screen

The analyzer screen features a graph of the input signals, both left and right channels, in a time domain display at the top of the screen, and in the frequency domain (FFT) at the bottom of the screen.

Kit Operation

Next to the time domain plots, calculated values of SINAD, SNR, and THD are shown. These values are all expressed in dB relative to the full-scale of the TLV320AIC32. Note that the SNR number shown is A-weighted.

In [Figure 4](#), a sine wave generated by the TLV320AIC32 DACs is fed through the high power drivers and back into the ADC; then the resulting FFTs can be seen. Note that this sequence is a full analog loopback test case, so the measured numbers show the combined performance of the DAC, drivers, and ADC. There is also no post-DAC filtering; as noted in the data sheet, this may degrade measurements even though there is no audible noise.

6.6 Audio Input/ADC Tab

The Audio Input/ADC Tab is laid out like an audio mixing console. Each input channel has a vertical strip that corresponds to that channel. *LINE1L* and *LINE1R* input strips have controls to route that input to either the left or right ADC input; by default, all inputs are muted when the TLV320AIC32 is powered up. To route an input to the ADC, first click on the **MUTE** button in the input channel strip which corresponds to the ADC input channel you want that input to go to—the caption on the button will change to *ACTIVE*. The level of the input channel routed to that particular ADC channel can then be adjusted using the *Level* knob below the **MUTE/ACTIVE** button. See [Figure 5](#).

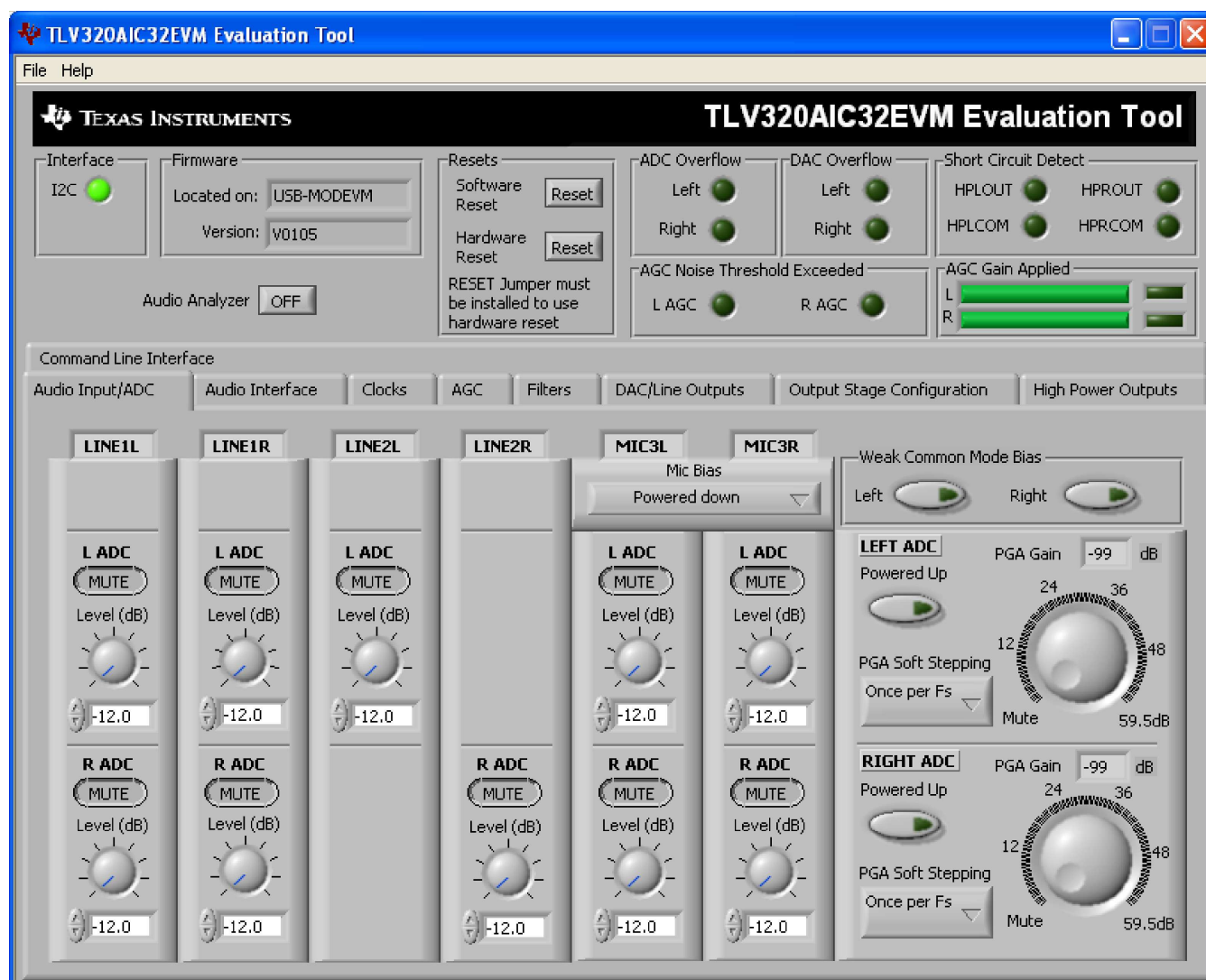


Figure 5. Audio Input Tab

Similarly, the *LINE2L* and *LINE2R* inputs can be routed to the ADC. However, note that these inputs do not have the choice to route to either ADC channel, they can only be configured to connect to the corresponding ADC input.

The *MIC3L* and *MIC3R* inputs are similar to the *LINE1L* and *LINE1R* inputs in that they can be routed to either ADC input channel. Control of the mic bias is accomplished by using the pull-down menu at the top of these channel strips. The mic bias can either be powered down or set to 2.0V, 2.5V, or the power supply voltage of the ADC (AVDD_ADC).

To use the on-board microphone, JMP3 and JMP4 must be installed and nothing should be plugged into J8. In order for the mic bias settings in the software to take effect, JMP10 should be set to connect positions 2 and 3, so that mic bias is controlled by the TLV320AIC32.

In the upper right of this tab are controls for **Weak Common Mode Bias**. Enabling these controls will result in unselected inputs to the ADC channels to be weakly biased to the ADC common mode voltage.

Below these controls are the controls for the ADC PGA—the master volume controls for the ADC inputs. Each channel of the ADC can be powered up or down as needed using the **Powered Up** buttons. PGA soft-stepping for each channel is selected using the control below this. The large knobs set the actual **ADC PGA Gain**; at the extreme counterclockwise rotation, the channel is muted. Rotating the knob clockwise increases the PGA gain.

6.7 Audio Interface Tab

The Audio Interface tab ([Figure 6](#)) sets up the audio data interface to the TLV320AIC32. For use with the PC software and the USB-MODEVM, the default settings should be used. If using an external I²S source, or other data source, the interface mode may be selected using the **Transfer Mode** control—selecting either I²S mode, DSP mode, or Right- or Left-Justified modes. Word length can be selected using the **Word Length** control, and the bit clock rate can also be selected using the **Bit Clock** rate control. The **Data Word Offset**, used in TDM mode (see the [product datasheet](#)) can also be selected on this tab.

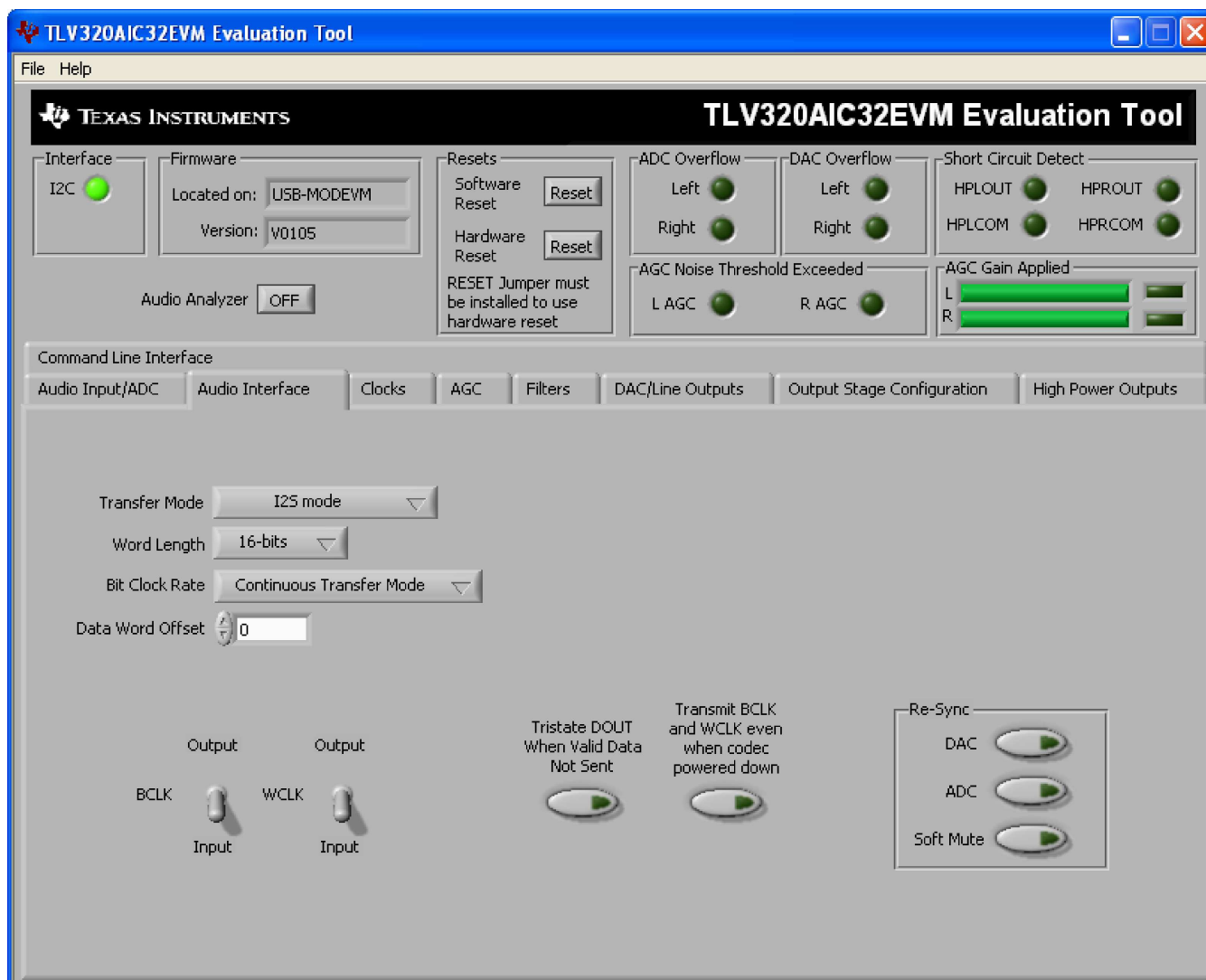


Figure 6. Audio Interface Tab

Along the bottom of this tab are controls for choosing the **BLCK** and **WCLK** as being either inputs or outputs, as well as options for tristating the DOUT line when there is not valid data and transmitting BLCK and WCLK when the codec is powered down.

Re-sync of the audio bus is enabled using the controls in the lower right corner of this screen. Re-sync is done if the group delay changes by more than $\pm FS/4$ for the ADC or DAC sample rates (see the [TLV320AIC32](#) datasheet). The channels can be soft muted when doing the re-sync if the **Soft Mute** button is enabled.

6.8 Clocks Tab

The TLV320AIC32 has a very flexible scheme for generating the clock sources for ADC and DAC sample rates. The Clocks tab allows access to set the different options for setting up these clocks. Refer to the Audio Clock Generation Processing figure (Figure 24) in the [TLV320AIC32](#) datasheet.

For use with the PC software and the USB-MODEVM, the clock settings must be set a certain way. These settings are not the default settings of the TLV320AIC32. The EVM-required settings can be loaded automatically by pushing the **Load EVM Clock Settings** button at the bottom of this tab. Note that changing any of the clock settings from the values loaded when this button is pushed may result in the EVM not working properly with the PC software or USB interface. If an external audio bus is used (audio not driven over the USB bus), then settings may be changed to any valid combination. See [Figure 7](#).

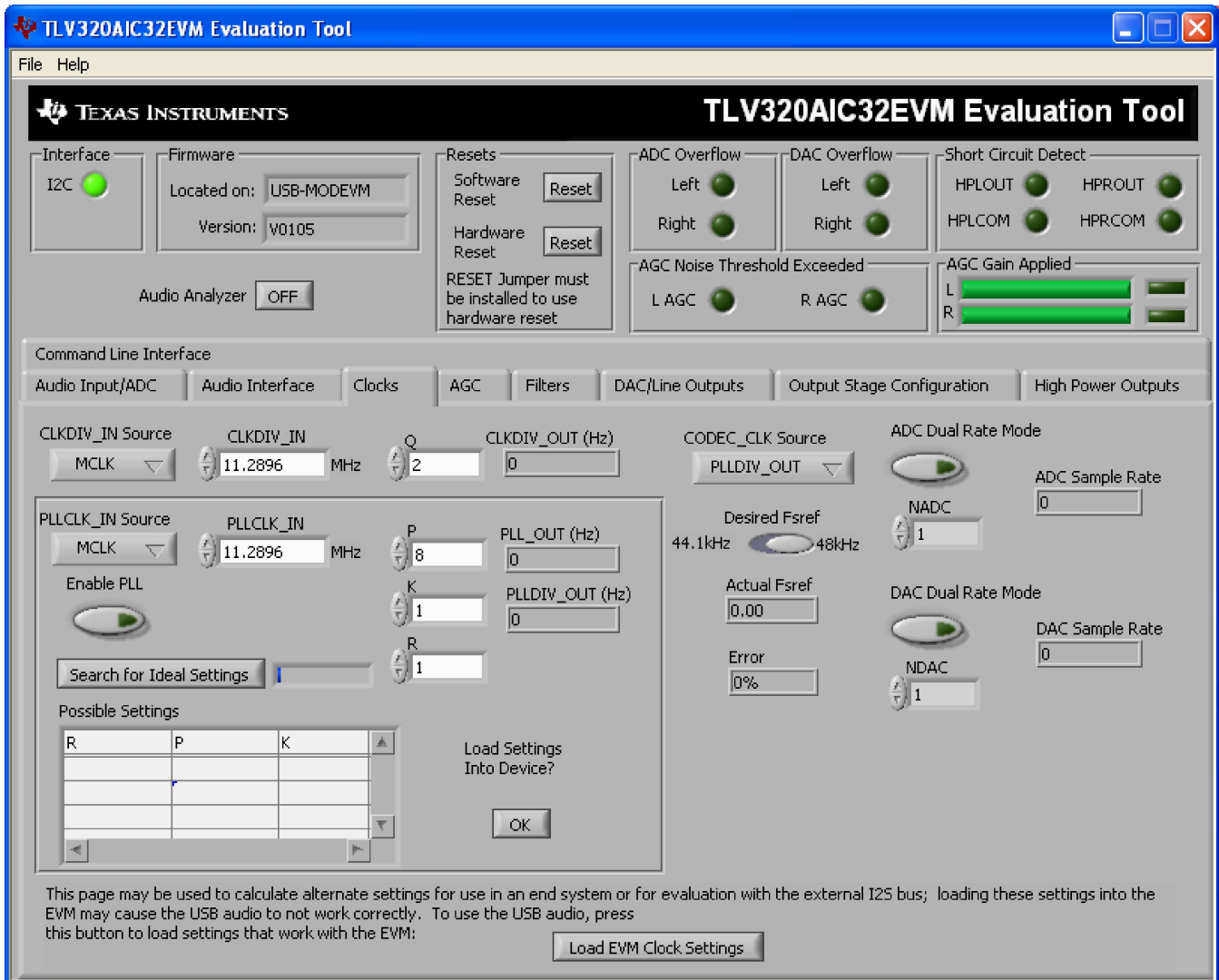


Figure 7. Clocks Tab

The codec clock source is chosen by the **CODEC_CLK Source** control. When this control is set to *CLKDIV_OUT*, the PLL is not used; when set to *PLLDIV_OUT*, the PLL is used to generate the clocks.

6.8.1 Use Without PLL

Setting up the TLV320AIC32 for clocking without using the PLL is straightforward. The **CLKDIV_IN** source can be selected as either *MCLK* or *BCLK*; the default is *MCLK*. The **CLKDIV_IN** frequency is then entered into the **CLKDIV_IN** box, in megahertz (MHz). The default value shown, 11.2896MHz, is the frequency used on the USB-MODEVM board. This value is then divided by the value of **Q**, which can be set from 2 to 17; the resulting **CLKDIV_OUT** frequency is shown in the indicator next to the **Q** control.

This frequency will then be used to calculate the actual *Fsref* frequency, and the ADC and DAC sample rates, after the **NADC** and **NDAC** factors are applied to the *Fsref*. If dual rate mode is desired, this option can be enabled for either the ADC or DAC by pressing the corresponding **Dual Rate Mode** button.

6.8.2 Use With The PLL

When **PLLDIV_OUT** is selected as the codec clock source, the PLL is used. The PLL clock source is chosen using the **PLLCLK_IN** control, and may be set to either *MCLK* or *BCLK*. The **PLLCLK_IN** frequency is then entered into the **PLLCLK_IN Source** box.

The **PLL_OUT** and **PLLDIV_OUT** indicators show the resulting PLL output frequencies with the values set for the **P**, **K**, and **R** parameters of the PLL. Refer to the [TLV320AIC32](#) datasheet for an explanation of these parameters. The parameters can be set by clicking on the up/down arrows of the **P**, **K**, and **R** combo boxes, or they can be typed into these boxes. The values can also be calculated by the PC software.

To use the PC software to find the ideal values of **P**, **K**, and **R** for a given PLL input frequency and desired *Fsref*, the desired **Fsref** must be set using the switch on this tab; it can be set to either 44.1kHz or 48kHz. Once the desired *Fsref* and **PLLCLK_IN** values are correctly set, pushing the **Search for Ideal Settings** button starts the software searching for ideal combinations of **P**, **K**, and **R** which achieve the desired *Fsref*. The possible settings for these parameters are displayed in the spreadsheet-like table labeled *Possible Settings*. Clicking on a row in this table sets the **P**, **K**, and **R** values in the software and updates the **PLL_OUT** and **PLLDIV_OUT** readings, as well as the *Actual Fsref* and *Error* displays. This process does not actually load the values into the TLV320AIC32, however; it only updates the displays in the software. This allows for different possible solutions to be selected and the error evaluated before loading into the device.

When a suitable combination of **P**, **K**, and **R** have been chosen, pressing the **Load Settings into Device?** button will download these values into the appropriate registers on the TLV320AIC32.

6.9 AGC Tab

The AGC tab (see [Figure 8](#)) consists of two identical sets of controls, one for the left channel and the other for the right channel. The AGC function is described in the [TLV320AIC32](#) datasheet.

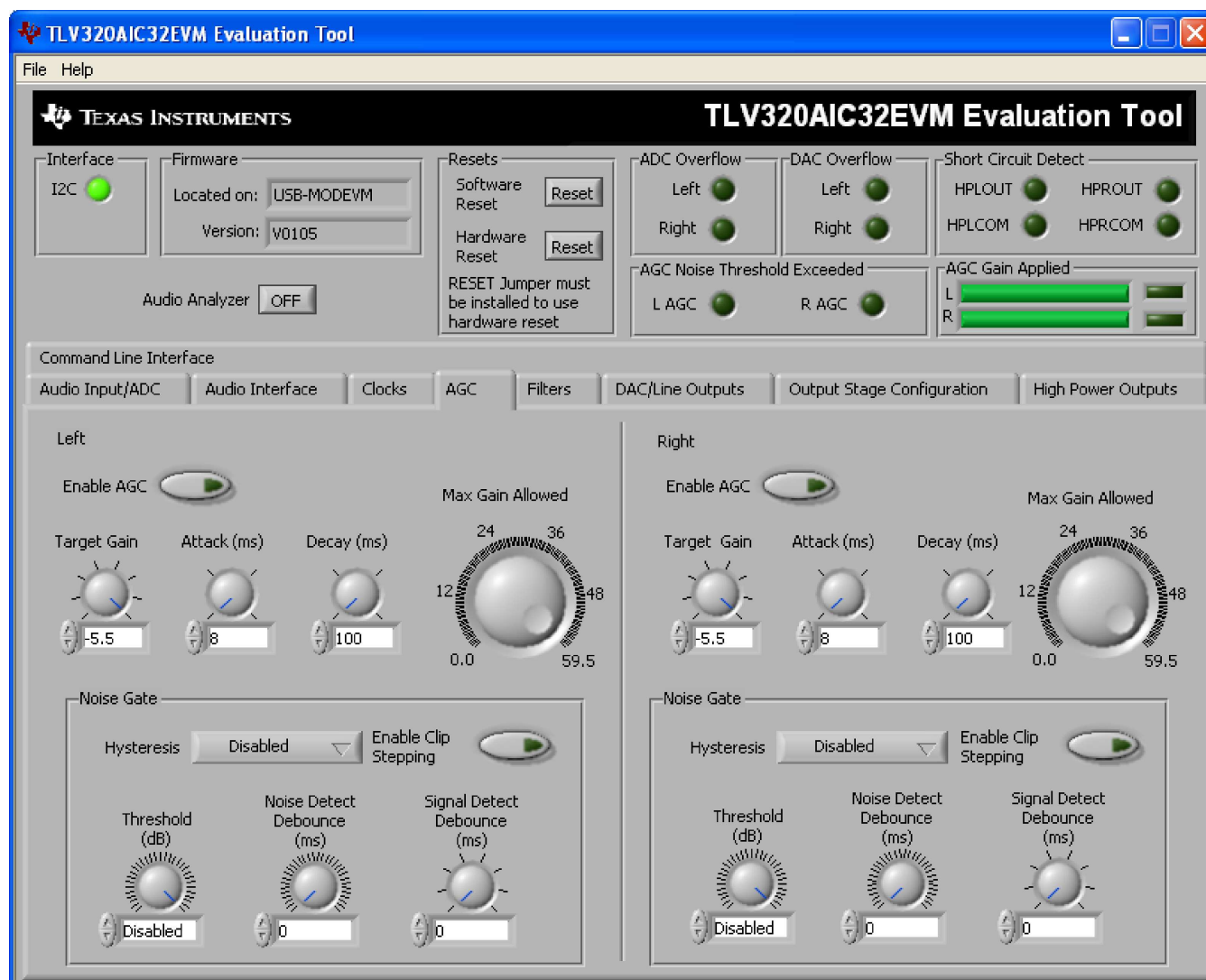


Figure 8. AGC Tab

The AGC can be enabled for each channel using the **Enable AGC** button. *Target* gain, *Attack* time in milliseconds, *Decay* time in milliseconds, and the *Maximum PGA Gain Allowed* can all be set, respectively, using the four corresponding knobs in each channel.

Noise gate functions, such as *Hysteresis*, *Clip stepping*, *Threshold*, and *Signal and Noise Detect* debouncing are set using the corresponding controls in the **Noise Gate** groupbox for each channel.

6.10 Filters Tab

The TLV320AIC32 has a very rich feature set for applying digital filtering to audio signals. This tab controls all of the filter features of the TLV320AIC32. In order to use this tab and plot filter responses correctly, the DAC sample rate must be set properly. Therefore, the clocks must be set up correctly in the software following the discussion in [Section 6.8](#). See [Figure 9](#).

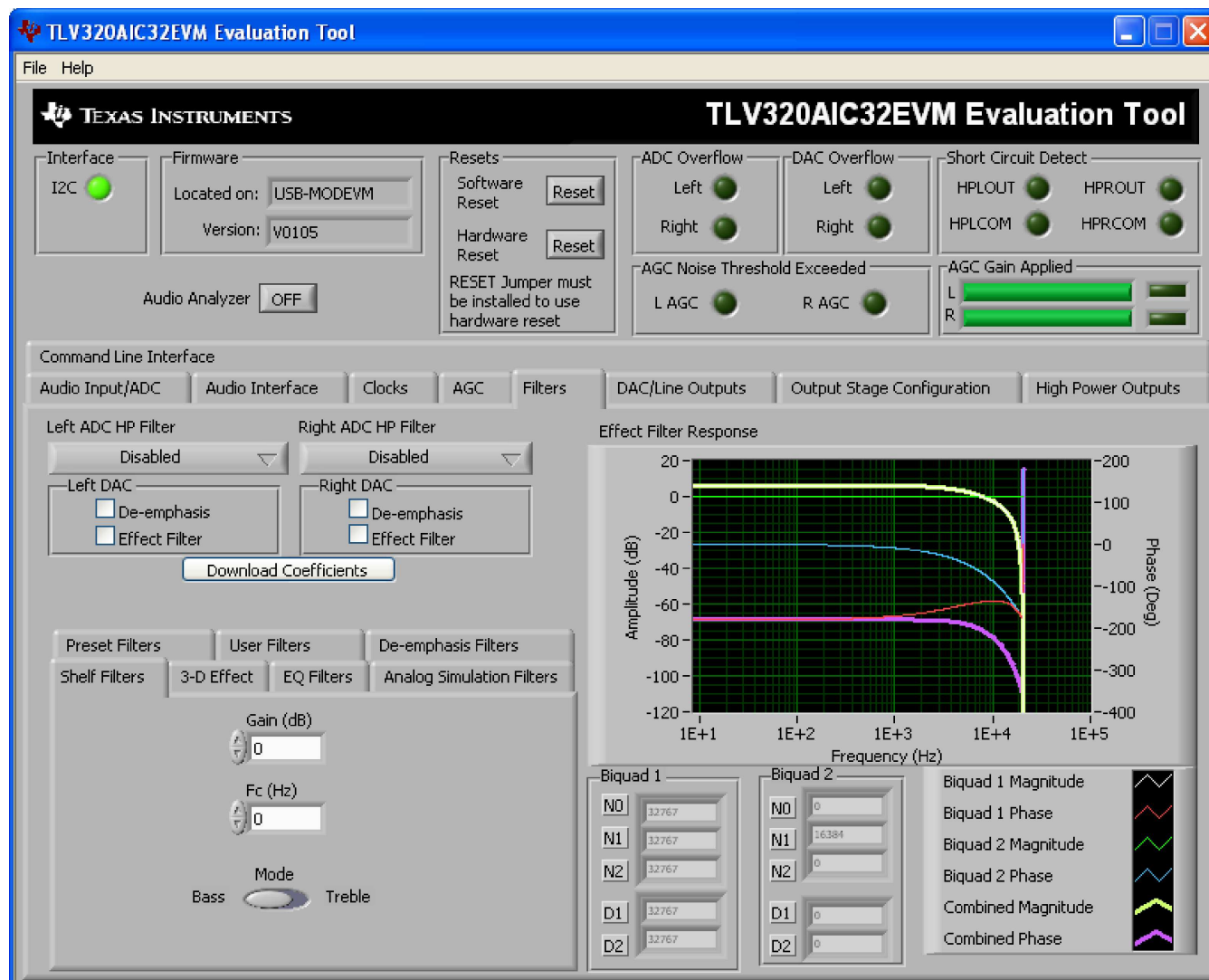


Figure 9. Filters Tab

The right-hand side of this tab shows a display which plots the magnitude and phase response of each biquad section, plus the combined responses of the two biquad sections. The coefficients used for the plotted responses are shown below the graph for both Biquad 1 and Biquad 2. Note that the plot shows only the responses of the effect filters, not the combined response of those filters along with the de-emphasis and ADC high-pass filters.

6.10.1 ADC Highpass Filters

The ADC of the TLV320AIC32 can have a high-pass filter enabled, which helps to reduce the effects of DC offsets in the system. This function is enabled as shown in Figure 10. The four options for this setting are disabled, or three different corner frequencies which are based on the ADC sample rate.

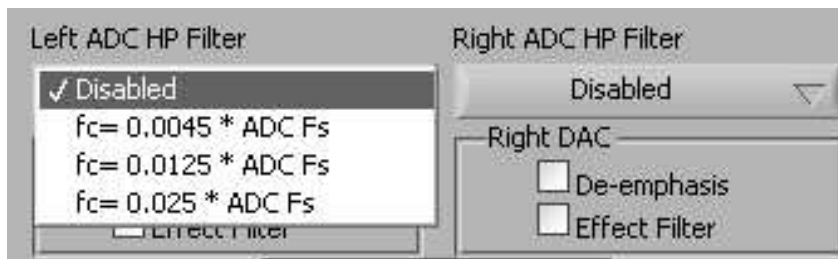


Figure 10. ADC Highpass Filter Settings

6.10.2 Enabling Filters

The de-emphasis and effect filters (the biquad filters) of the TLV320AIC32 are selected using the checkboxes shown in Figure 11. The De-emphasis filters are described in the [TLV320AIC32](#) datasheet, and their coefficients may be changed (see Section 6.10.7).

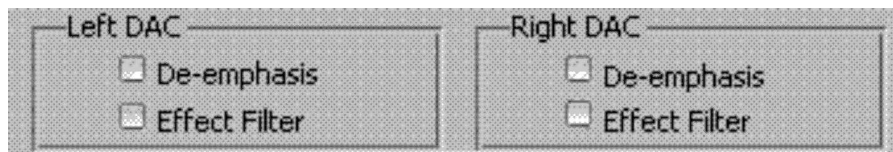


Figure 11. Enabling Filters

When designing filters for use with TLV320AIC32, the software allows for several different filter types to be used. These options are shown on a tab control in the lower left corner of the screen. When a filter type is selected, and suitable input parameters defined, the response will be shown in the *Effect Filter Response* graph. Regardless of the setting for enabling the Effect Filter, the filter coefficients are not loaded into the TLV320AIC32 until the **Download Coefficients** button is pressed. To avoid noise during the update of coefficients, it is recommended that you uncheck the **Effect Filter enable** checkboxes before downloading coefficients. Once the desired coefficients are in the TLV320AIC32, enable the Effect Filters by checking the boxes again.

6.10.3 Shelf Filters

A shelf filter is a simple filter that applies a gain (positive or negative) to frequencies above or below a certain corner frequency. As shown in Figure 12, in *Bass* mode, a shelf filter applies a gain to frequencies below the corner frequency; in *Treble* mode, the gain is applied to frequencies above the corner frequency.

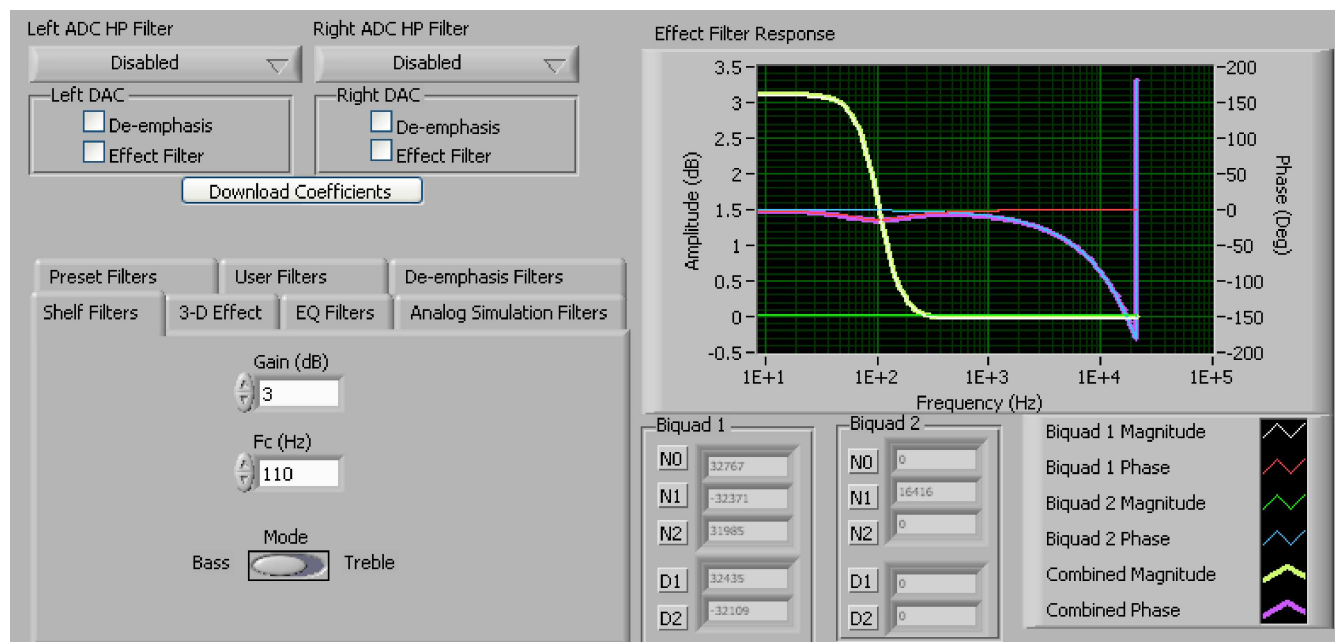


Figure 12. Shelf Filters

To use these filters, enter the gain desired and the corner frequency. Choose the mode to use (*Bass* or *Treble*); the response will be plotted on the *Effect Filter Response* graph.

6.10.4 EQ Filters

EQ, or parametric, filters can be designed on this tab. Enter a gain, bandwidth, and a center frequency (Fc). Either bandpass (positive gain) or band-reject (negative gain) filters can be created.

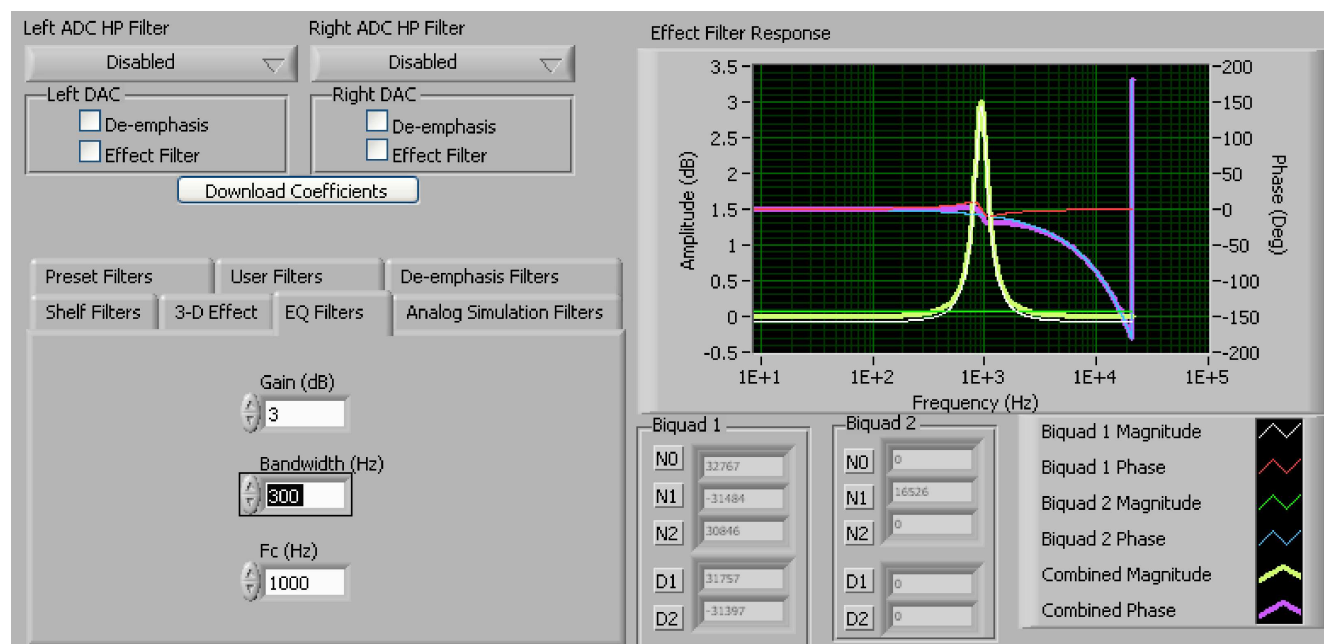


Figure 13. EQ Filters

6.10.5 Analog Simulation Filters

Biquads are quite good at simulating analog filter designs. For each biquad section on this tab, enter the desired analog filter type to simulate (Butterworth, Chebyshev, Inverse Chebyshev, Elliptic or Bessel). Parameter entry boxes appropriate to the filter type will be shown (ripple, for example, with Chebyshev filters, etc.). Enter the desired design parameters and the response will be shown.

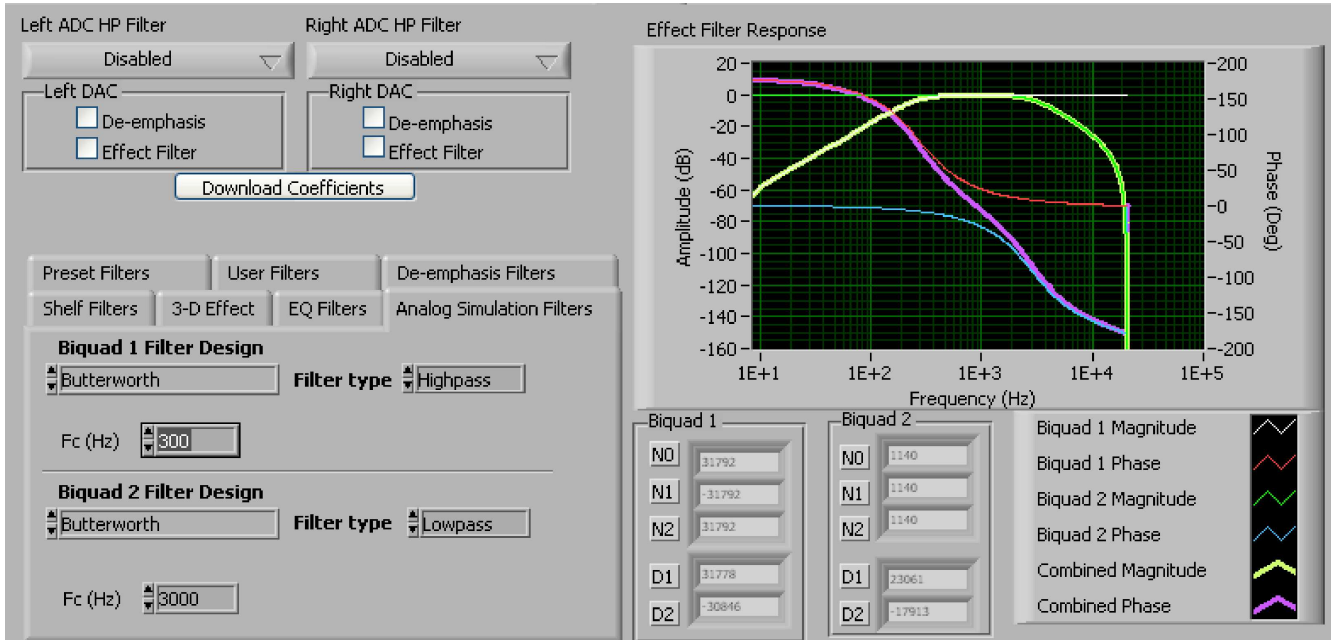


Figure 14. Analog Simulation Filters

6.10.6 Preset Filters

Many applications are designed to provide preset filters common for certain types of program material. This tab allows selection of one of four preset filter responses - Rock, Jazz, Classical, or Pop.

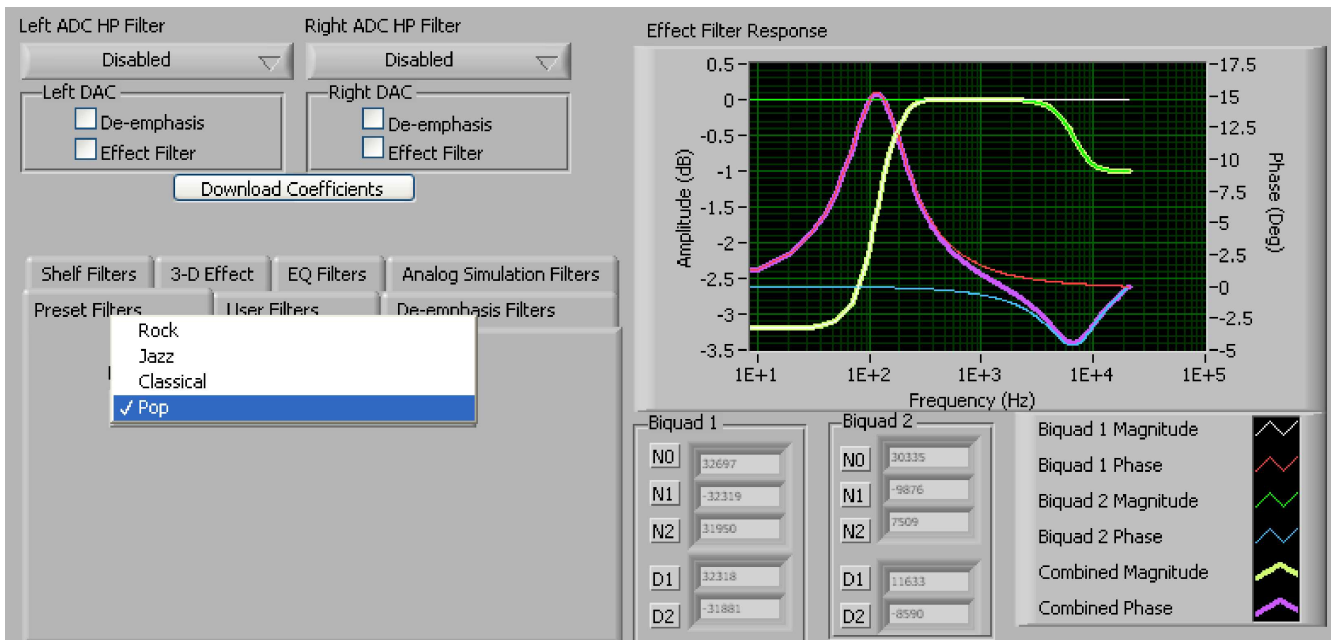


Figure 15. Preset Filters

6.10.7 De-emphasis Filters

The de-emphasis filters used in the TLV320AIC32 can be programmed as described in the [TLV320AIC32](#) datasheet, using this tab. Enter the coefficients for the de-emphasis filter response desired. While on this tab, the de-emphasis response will be shown on the *Effect Filter Response* graph; however, note that this response is not included in graphs of other effect responses when on the other filter design tabs.

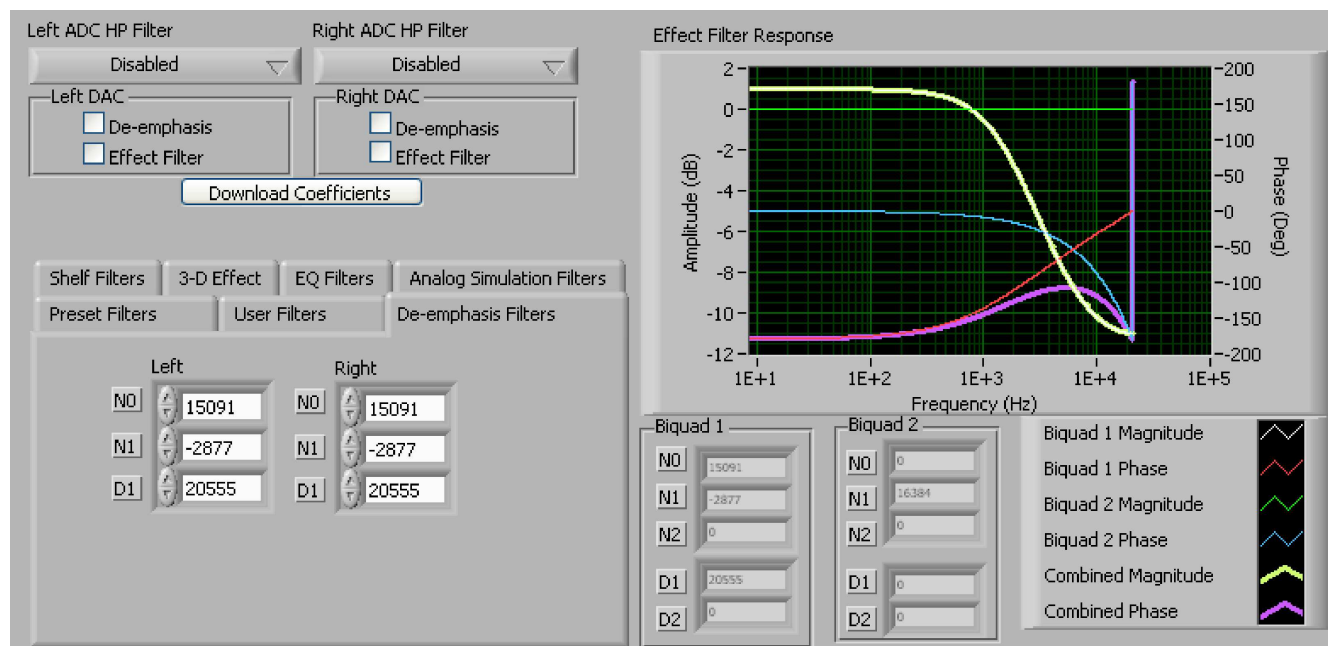


Figure 16. De-emphasis Filters

6.10.8 User Filters

If filter coefficients are known, they can be entered directly on this tab (see [Figure 17](#)) for both biquads for both left and right channels. The filter response will **not** be shown on the *Effect Filter Response* graph for user filters.

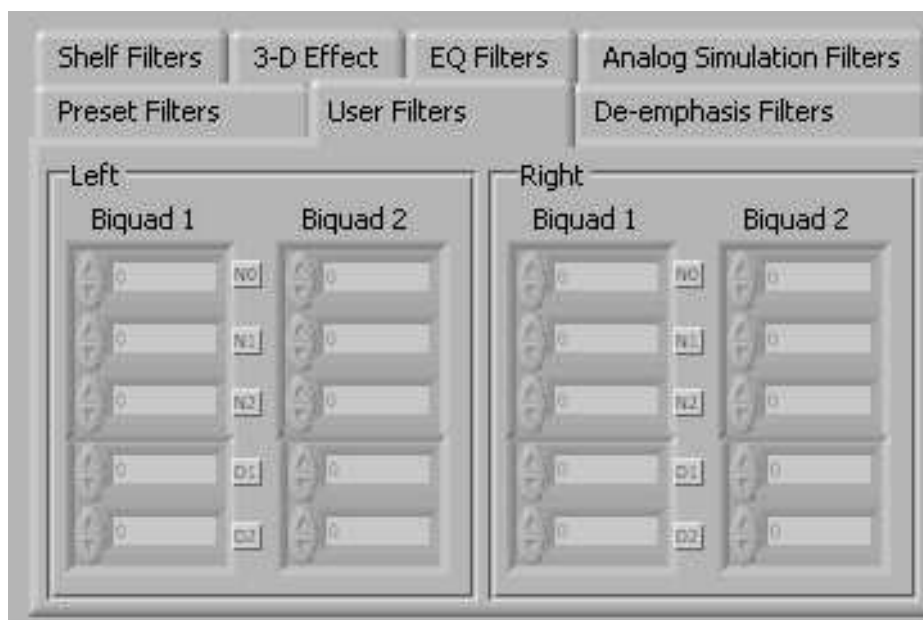


Figure 17. User Filters

6.10.9 3D Effect

The 3D effect is described in the [TLV320AIC32](#) datasheet. It uses the two biquad sections differently than most other effect filter settings. To use this effect properly, make sure the appropriate coefficients are already loaded into the two biquad sections. The User Filters tab may be used to load the coefficients. See [Figure 18](#).

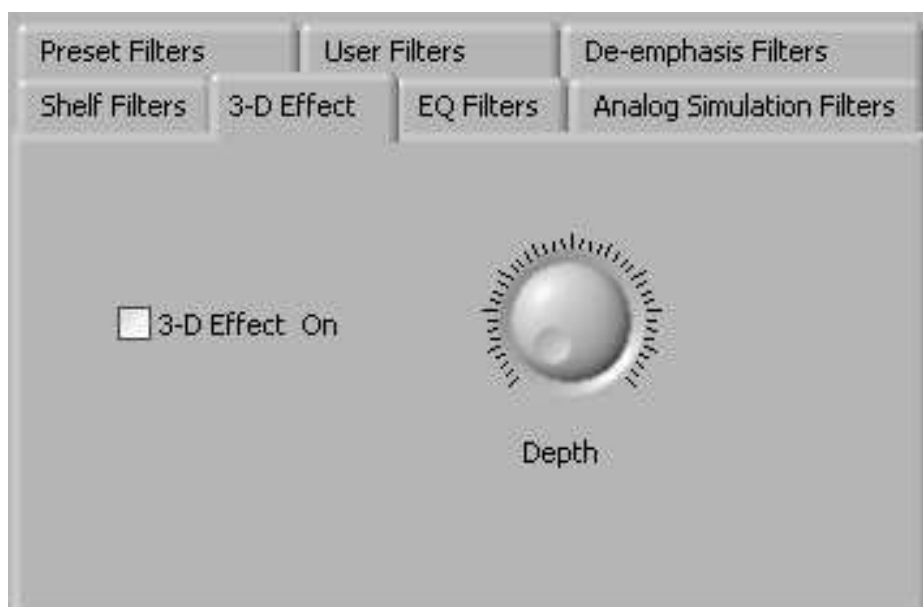


Figure 18. 3D Effect Settings

Kit Operation

To enable the 3D effect, check the *3D Effect On* box. The **Depth** knob controls the value of the 3D Attenuation Coefficient.

6.11 DAC/Line Outputs Tab

The DAC/Line Outputs tab controls the DAC power and volume, as well as routing of digital data to the DACs and the analog output from the DACs. (See [Figure 19](#).)

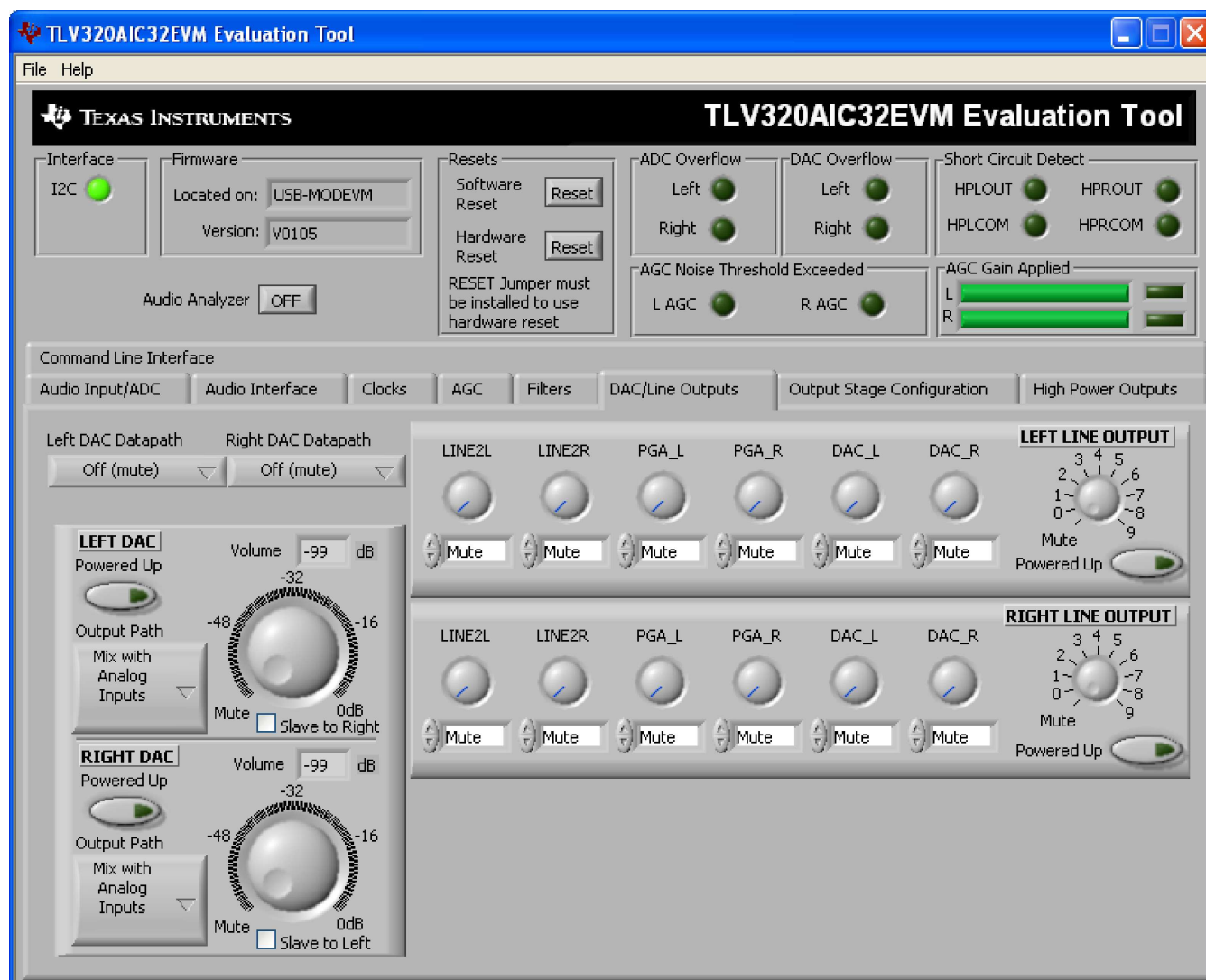


Figure 19. DAC/Line Outputs Tab

6.11.1 DAC Controls

On the left side of this tab are controls for the left and right DACs.

In a similar fashion as the ADC, the DAC controls are set up to allow powering of each DAC individually, and setting the output level. Each channel level can be set independently using the corresponding **Volume** knob. Alternately, by checking the **Slave to Right** box, the left channel Volume can be made to track the right channel Volume knob setting; checking the **Slave to Left** box causes the right channel Volume knob to track the left Volume knob setting.

Data going to the DACs is selected using the drop-down boxes under the **Left and Right Datapath**. Each DAC channel can be selected to be off, use left channel data, use right channel data, or use a mono mix of the left and right data.

Analog audio coming from the DACs is routed to outputs using the **Output Path** controls in each DAC control panel. The DAC output can be mixed with the analog inputs (LINE2L, LINE2R, PGA_L, PGA_R) and routed to the Line or High Power outputs using the mixer controls for these outputs on this tab (for the line outputs) or on the High Power Outputs tab (for the high power outputs). If the DAC is to be routed directly to either the Line or HP outputs, these routes can be selected as choices in the Output Path control. Note that if the Line or HP outputs are selected as the Output Path, the mixer controls on this tab and the High Power Output tabs have no effect.

6.11.2 Line Output Mixers

On the right side of this tab are horizontal panels that house the mixing functions for the line outputs.

Each line output master volume is controlled by the knob at the far right of these panels. The output can be muted, or gain up to 9dB can be applied. Power for the line output can also be controlled using the button below this master output knob.

If the DAC output path is set to *Mix with Analog Inputs*, the six knobs in each panel can be used to set the individual level of signals routed and mixed to the line output. LINE2L, LINE2R, PGA_L, PGA_R, and DAC_L and DAC_R levels can each be set to create a custom mix of signals presented to that particular line output. **Note:** if the DAC output path is set to anything other than *Mix with Analog Inputs*, these controls have no effect.

6.12 Output Stage Configuration Tab

The Output Stage Configuration tab ([Figure 20](#)) allows for setting several features of the output drivers. The **Configuration** may be set as either *Fully-Differential* or *Pseudo-Differential*. The output coupling can be chosen as either capless or AC-coupled. This setting should correspond to the setting of the hardware switch (SW1) on the TLV320AIC32EVM.

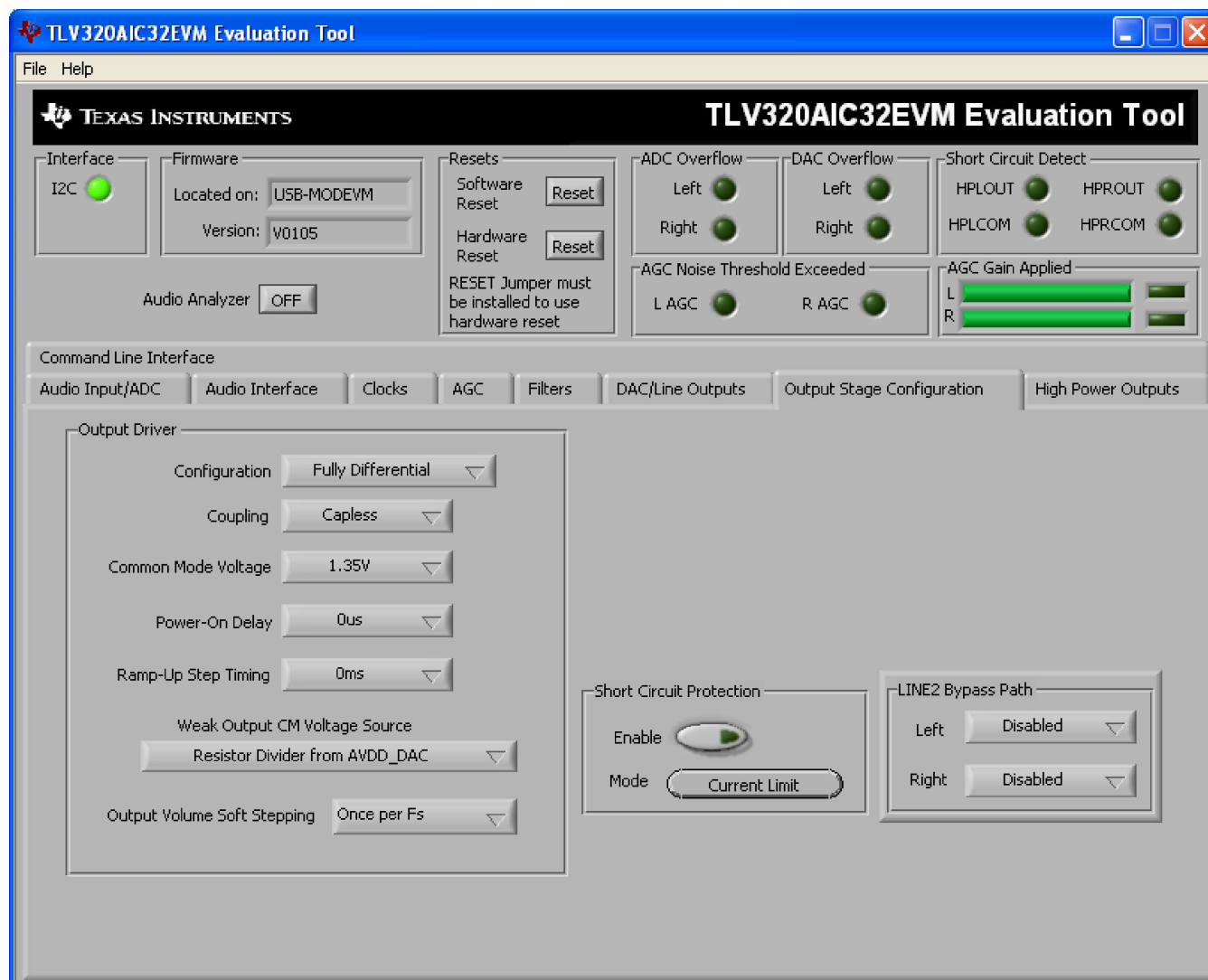


Figure 20. Output Stage Configuration Tab

The Common Mode Voltage of the outputs may be set to 1.35V, 1.5V, 1.65V, or 1.8V using the **Common Mode Voltage** control. The **Power-On Delay** of the output drivers can be set using the corresponding control from 0 μ s up to 4 seconds. **Ramp-Up Step Timing** can also be adjusted from 0ms to 4ms.

The high power outputs of the TLV320AIC32 can be configured to go to a weak common-mode voltage when powered down. The source of this weak common-mode voltage can be set on this tab with the **Weak Output CM Voltage Source** drop-down. Choices for the source are either a resistor divider off the AVDD_DAC supply, or a bandgap reference. See the [TLV320AIC32](#) datasheet for more details on this option.

The outputs can be set to soft-step their volume changes, using the **Output Volume Soft Stepping** control, and set to step once per Fs period, once per two Fs periods, or soft stepping can be disabled altogether.

Output short-circuit protection can be enabled in the **Short Circuit Protection** groupbox. Short Circuit Protection can use a current-limit mode, where the drivers will limit current output if a short-circuit condition is detected, or in a mode where the drivers power down when such a condition exists.

The LINE2 Bypass Path groupbox has controls that allow disabling or routing the LINE2 input to the output stage directly.

6.13 High Power Outputs Tab

This tab contains four horizontal groupings of controls, one for each of the high power outputs. Each output has a mixer to mix the LINE2L, LINE2R, PGA_L, PGA_R, DAC_L and DAC_R signals, assuming that the DACs are not routed directly to the high power outputs (see Figure 21).

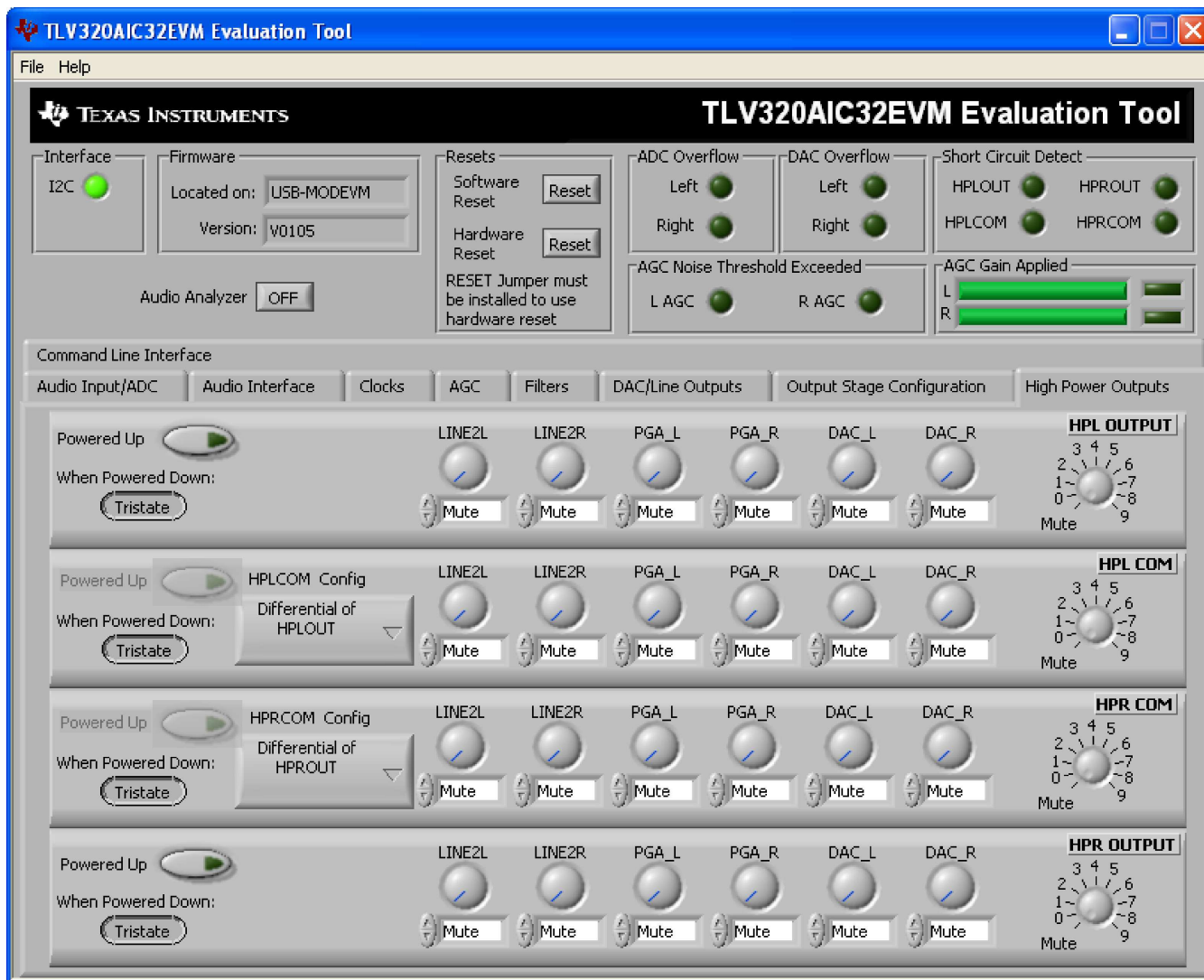


Figure 21. High Power Outputs Tab

At the left of each output strip is a power button that controls whether the corresponding output is powered up or not. When powered down, the outputs can be tri-stated or driven weakly to the output common mode voltage; this option is selected using the button located below the power button.

The **COM** outputs (*HPLCOM* and *HPRCOM*) can be used as independent output channels or can be used as complementary signals to the HPL and HPR outputs. In these complementary configurations, the COM outputs can be selected as differential signals to the corresponding outputs or may be set to be a common mode voltage. When used in these configurations, the power button for the COM output is disabled, as the power mode for that output will track the power status of the HPL or HPR output that the COM output is tracking.

At the right side of the output strip is a master volume knob for that output, which allows muting the output or applying gain up to 9dB.

6.14 Command Line Interface Tab

A simple scripting language controls the TAS1020 on the USB-MODEVM from the LabView™-based PC software. The main program controls, described previously, do nothing more than write a script which is then handed off to an interpreter that sends the appropriate data to the correct USB endpoint. Because this system is script-based, provision is made in this tab for the user to view the scripting commands that are created as the controls are manipulated, as well as load and execute other scripts that have been written and saved (see Figure 22). This design allows the software to be used as a quick test tool or to help provide troubleshooting information in the rare event that the user encounters a problem with this EVM.

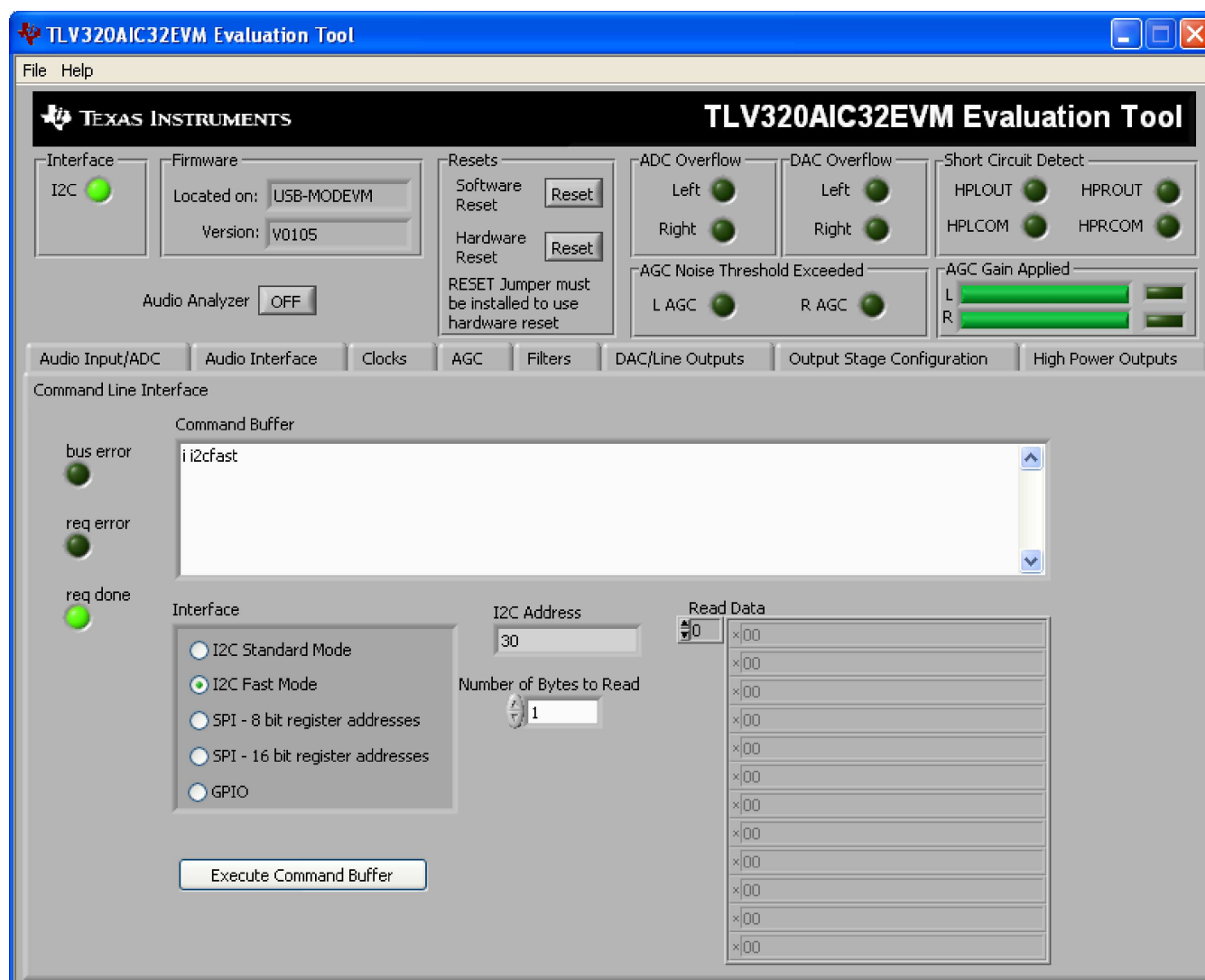


Figure 22. Command Line Interface Tab

A script is loaded into the command buffer, either by operating the controls on the other tabs or by loading a script file. When executed, the return packets of data which result from each command will be displayed in the **Read Data** array control. When executing several commands, the Read Data control shows only the results of the last command. If you want to see the results after every executed command, use the logging function described below.

The File menu (Figure 23) provides some options for working with scripts. The first option, *Open Command File...*, loads a command file script into the command buffer. This script can then be executed by pressing the **Execute Command Buffer** button.

The second option is *Log Script and Results...*, which opens a file save dialog box. Choose a location for a log file to be written using this file save dialog. When the Execute Command Buffer button is pressed, the script will run and the script, along with resulting data read back during the script, will be saved to the file specified. The log file is a standard text file which can be opened with any text editor, and looks much like the source script file, but with the additional information of the result of each script command executed.

The third menu item is a submenu of *Recently Opened Files*. This list is simply a list of script files that have previously been opened, allowing fast access to commonly-used script files. The final menu item is *Exit*, which terminates the TLV320AIC32EVM software.

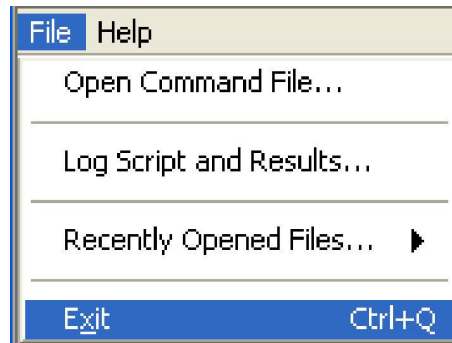


Figure 23. File Menu

Under the Help menu is an *About...* menu item which displays information about the TLV320AIC32EVM software.

The actual USB protocol used as well as instructions on writing scripts are detailed in the following subsections. While it is not necessary to understand or use either the protocol or the scripts directly, understanding them may be helpful to some users.

6.14.1 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 TAS1020 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 passes from the PC to the TAS1020 using the control endpoint.

Data is sent in an HIDSETREPORT (see Table 7):

**Table 7. 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 below

The data packet consists of the following bytes, shown in [Table 8](#):

Table 8. Data Packet Configuration

BYTE NUMBER	TYPE	DESCRIPTION														
0	Interface	Specifies serial interface and operation. The two values are logically OR'd. Operation: <table style="margin-left: 40px;"> <tr> <td>READ</td> <td>0x00</td> </tr> <tr> <td>WRITE</td> <td>0x10</td> </tr> </table> Interface: <table style="margin-left: 40px;"> <tr> <td>GPIO</td> <td>0x08</td> </tr> <tr> <td>SPI_16</td> <td>0x04</td> </tr> <tr> <td>I2C_FAST</td> <td>0x02</td> </tr> <tr> <td>I2C_STD</td> <td>0x01</td> </tr> <tr> <td>SPI_8</td> <td>0x00</td> </tr> </table>	READ	0x00	WRITE	0x10	GPIO	0x08	SPI_16	0x04	I2C_FAST	0x02	I2C_STD	0x01	SPI_8	0x00
READ	0x00															
WRITE	0x10															
GPIO	0x08															
SPI_16	0x04															
I2C_FAST	0x02															
I2C_STD	0x01															
SPI_8	0x00															
1	I ² C Slave Address	Slave address of I ² C 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 ² C 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 I²C 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 let's 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 will return, 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] should be the same as the sent one logically or'd with 0x20 - in our first example above, the returning packet should be:

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

If for some reason the interface fails (for example, the I²C device does not acknowledge), it would come 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 described above, the return packet would be:

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

Examples above used writes. Reading is similar:

Read two bytes from device starting at register 5 of an I²C device with address A0:

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

The return packet should be

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

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

6.14.1.1 GPIO Capability

The USB-MODEVM has seven GPIO lines. You can 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 9](#)):

Table 9. GPIO Pin Assignments

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
```

You may also read back from the GPIO to see the state of the pins. Let's say we just wrote the previous example 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 should be:

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

6.14.2 Writing Scripts

A script is simply a text file that contains data to send to the serial control buses. The scripting language is quite simple, as is the parser for the language. Therefore, the program is not very forgiving about mistakes made in the source script file, but the formatting of the file is simple. Consequently, mistakes should be rare.

Each line in a script file is one command. There is no provision for extending lines beyond one line. 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
- #** Comment
- b** Break
- d** Delay

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 ² C Bus
i2cfast	Fast mode I ² C 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 I²C bus is to be used, the script would begin with:

i i2cfast

No data follows the break command. Anything following a comment command is ignored by the parser, provided that it is on the same line. The delay command allows the user to specify a time, in milliseconds, that the script will pause before proceeding.

Note: UNLIKE ALL OTHER NUMBERS USED IN THE SCRIPT COMMANDS, THE DELAY TIME IS ENTERED IN A DECIMAL FORMAT. Also, note that because of latency in the USB bus as well as the time it takes the processor on the USB-MODEVM to handle requests, the delay time may not be precise.

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 TAS1020 by the program using the protocol described in [Section 6.14.1](#).

The first byte following a read or write command is the I²C slave address of the device (if I²C 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 will vary 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 I²C; SPI varies—see [Section 6.14.1](#) 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 I²C device with a slave address of 0x90, starting at a register address of 0x03, one would write:

```
#example script
i i2cfast
w 90 03 AA 55
r 90 03 2
```

This script begins with a comment, specifies that a fast I²C bus will be used, then writes 0xAA 0x55 to the I²C slave device at address 0x90, 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 not necessary to set the R/W bit for I²C devices in the script; the read or write commands will do that for you.

Here is an example of using an SPI device that requires 16-bit register addresses:

```
# setup TSC2101 for input and output
# uses SPI16 interface
# this script sets up DAC and ADC at full volume, input from onboard mic
#
# Page 2: Audio control registers
w 10 00 00 00 80 00 00 00 45 31 44 FD 40 00 31 C4
w 13 60 11 20 00 00 00 80 7F 00 C5 FE 31 40 7C 00 02 00 C4 00 00 00 23 10 FE 00 FE 00
```

Note that blank lines are allowed. However, be sure that your script does not end with a blank line. While ending with a blank line will not cause the script to fail, the program will execute that line, and therefore may prevent you from seeing data that was written or read back on the previous command.

In this example, the first two bytes of each command are the command word to send to the TSC2101 (0x1000, 0x1360); these are followed by data to write to the device starting at the address specified in the command word. The second line may wrap in the viewer you are using to look like more than one line; careful examination will show, however, that there is only one carriage return on that line, following the last **00**.

Any text editor may 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 Command File...* from the File menu. Locate your script and open it. The script will then be displayed in the command buffer. You may also edit the script once it is in the buffer, but saving of the command buffer is not possible at this time (this feature may be added at a later date).

Once the script is in the command buffer, it may be executed by pressing the *Execute Command Buffer* button. If you have placed breakpoints in your script, the script will execute to that point, and you will be presented with a dialog box with a button to press to continue executing the script. When you are ready to proceed, push that button and the script will continue.

Here an example of a (partial) script with breakpoints:

```
# setup AIC33 for input and output
# uses I2C interface
i i2cfast
# reg 07 - codec datapath
w 30 07 8A
r 30 07 1
d 1000
# regs 15/16 - ADC volume, unmute and set to 0dB
w 30 0F 00 00
r 30 0F 2
b
```

This script writes the value 8A at register 7, then reads it back to verify that the write was good. A delay of 1000ms (one second) is placed after the read to pause the script operation. When the script continues, the values **00 00** will be written starting at register 0F. This output is verified by reading two bytes, and pausing the script again, this time with a break. The script would not continue until the user allows it to by pressing *OK* in the dialog box that will be displayed due to the break.

7 EVM Bill of Materials

[Table 10](#) and [Table 11](#) contain a complete bill of materials for the modular TLV320AIC32EVM and the USB-MODEVM Interface Board (included only in the TLV320AIC32EVM-PDK).

Table 10. TLV320AIC32EVM Bill of Materials

REFERENCE DESIGNATOR	DESCRIPTION	MANUFACTURER	MFG PART NUMBER
R7, R8	0Ω 1/4W 5% chip resistor	Panasonic	ERJ-8GEY0R00V
R5, R6	2.2kΩ 1/4W 5% chip resistor	Panasonic	ERJ-8GEYJ222V
R1, R2, R3	2.7kΩ 1/10W 5% chip resistor	Panasonic	ERJ-3GEYJ272V
R9	100kΩ 1/10W 5% chip resistor	Panasonic	ERJ-3GEYJ104V
R4	Chip resistor	Not installed	
C5, C6, C9–C12	0.1μF 16V ceramic chip capacitor, ±10%, X7R	TDK	C1608X7R1C104K
C7–C8, C18–C19, C27–C28	0.1μF 100V ceramic chip capacitor, ±10%, X7R	TDK	C3216X7R2A104K
C1–C4, C13, C14, C20	10μF 6.3V ceramic chip capacitor, ±10%, X5R	TDK	C3216X5R0J106K
C21–C26	47μF 6.3V ceramic chip capacitor, ±20%, X5R	TDK	C3225X5R0J476M
C16, C17	Ceramic chip capacitor	Not installed	
C15	Ceramic chip capacitor	Not installed	
U3	Audio codec	Texas Instruments	TLV320AIC32IRHB
U1	3.3V LDO voltage regulator	Texas Instruments	REG1117-3.3
U2	64K I ² C EEPROM	MicroChip	24LC64-I/SN
J10, J11	Screw terminal block, 2-position	On Shore Technology	ED555/2DS
J6–J7, J12–J14	Screw terminal block, 3-position	On Shore Technology	ED555/3DS
J8, J9	3.5mm audio jack, T-R-S, SMD	CUI Inc.	SJ1-3515-SMT
	or alternate	KobiConn	161-3335
J1A, J2A, J4A, J5A	20-pin SMT plug	Samtec	TSM-110-01-L-DV-P
J1B, J2B, J4B, J5B	20-pin SMT socket	Samtec	SSW-110-22-F-D-VS-K
J3A	10-pin SMT plug	Samtec	TSM-105-01-L-DV-P
J3B	10-pin SMT socket	Samtec	SSW-105-22-F-D-VS-K
N/A	TLV320AIC32EVM PWB	Texas Instruments	6465230
JMP1–JMP4, JMP9, JMP11–JMP15	2-position jumper, 0.1" spacing	Samtec	TSW-102-07-L-S
JMP5–JMP8	Bus wire		
JMP10	3-position jumper, 0.1" spacing	Samtec	TSW-103-07-L-S
MK1	Omnidirectional microphone cartridge	Knowles Acoustics	MD9745APZ-F
SW1	4PDT right angle switch	E-Switch	EG4208
TP3–TP5, TP7–TP17, TP19–TP29	Miniature test point terminal	Keystone Electronics	5000
TP1, TP2	Multipurpose test point terminal	Keystone Electronics	5011
N/A	Header shorting block	Samtec	SNT-100-BK-T

Table 11. USB-MODEVM Bill of Materials

Designators	Description	Manufacturer	Mfg. Part Number
R4	10Ω 1/10W 5% chip resistor	Panasonic	ERJ-3GEYJ100V
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-3GEYJ152V
R1, R2, R3, R5, R6, R7, 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-3GEYJ103V
R17, R18	100kΩ 1/10W 5% chip resistor	Panasonic	ERJ-3GEYJ104V
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, C10, C11, C12, C22, C23, C24, C25, C26, C27, C28	1μF 6.3V ceramic chip capacitor, ±10%, X5R	TDK	C1608X5R0J105K
C1, C2, C3, C4, C5, C6, C7, C8	10μF 6.3V ceramic chip capacitor, ±10%, X5R	TDK	C3216X5R0J106K
D1	50V, 1A, Diode MELF SMD	Micro Commercial Components	DL4001
D2	Yellow Light Emitting Diode	Lumex	SML-LX0603YW-TR
D3, D4, D6, 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, tri-state buffers	Texas Instruments	SN74LVC125APW
U5, U6, U7	Single IC buffer driver with open drain o/p	Texas Instruments	SN74LVC1G07DBVR
U10	Single tri-state buffer	Texas Instruments	SN74LVC1G125DBVR
U1	64K 2-Wire serial EEPROM I ² C	Microchip	24LC64I/SN
	USB-MODEVM PCB	Texas Instruments	6463995

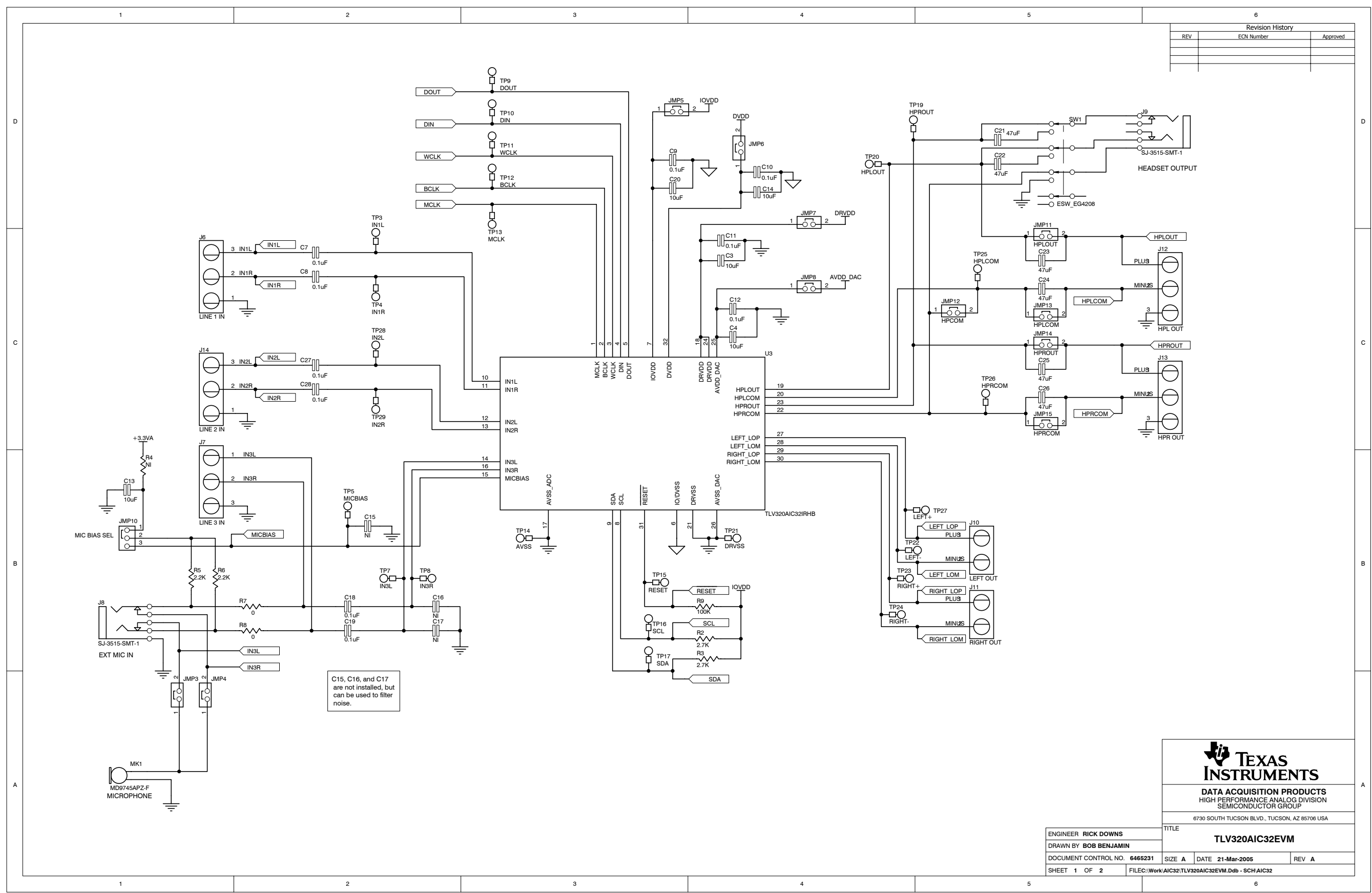
Table 11. USB-MODEVM Bill of Materials (continued)

Designators	Description	Manufacturer	Mfg. Part Number
TP1, TP2, TP3, TP4, TP5, TP6, TP9, TP10, 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
J1, J2, J3, J4, J5, J8	2-position terminal block	On Shore Technology	ED555/2DS
J9	2.5mm power connector	CUI Stack	PJ-102B
J10	BNC connector, female, PC mount	AMP/Tyco	414305-1
J11A, J12A, J21A, J22A	20-pin SMT plug	Samtec	TSM-110-01-L-DV-P
J11B, J12B, J21B, J22B	20-pin SMT socket	Samtec	SSW-110-22-F-D-VS-K
J13A, J23A	10-pin SMT plug	Samtec	TSM-105-01-L-DV-P
J13B, 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
J14, J15	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
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 A TLV320AIC32EVM Schematic

The schematic diagram is provided as a reference.

Revision History		
REV	ECN Number	Approved

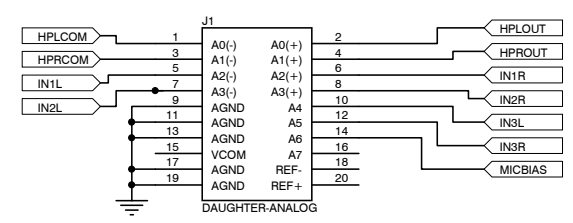


C15, C16, and C17 are not installed, but can be used to filter noise.

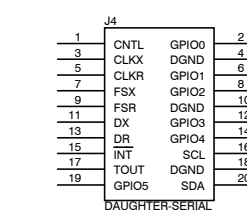


ENGINEER RICK DOWNS	TITLE		
DRAWN BY BOB BENJAMIN	TLV320AIC32EVM		
DOCUMENT CONTROL NO. 6465231	SIZE A	DATE 21-Mar-2005	REV A
SHEET 1 OF 2	FILEC:\Work\AIC32\TLV320AIC32EVM.Ddb - SCH\AIC32		

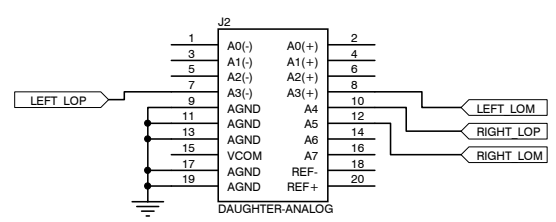
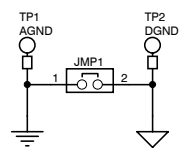
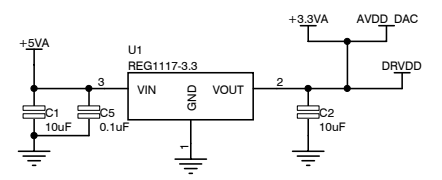
REVISION HISTORY		
REV	ENGINEERING CHANGE NUMBER	APPROVED



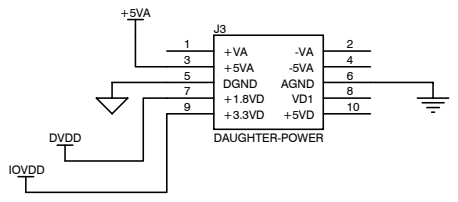
J1A (TOP) = SAM_TSM-110-01-L-DV-P
 J1B (BOTTOM) = SAM_SSW-110-22-F-D-VS-K



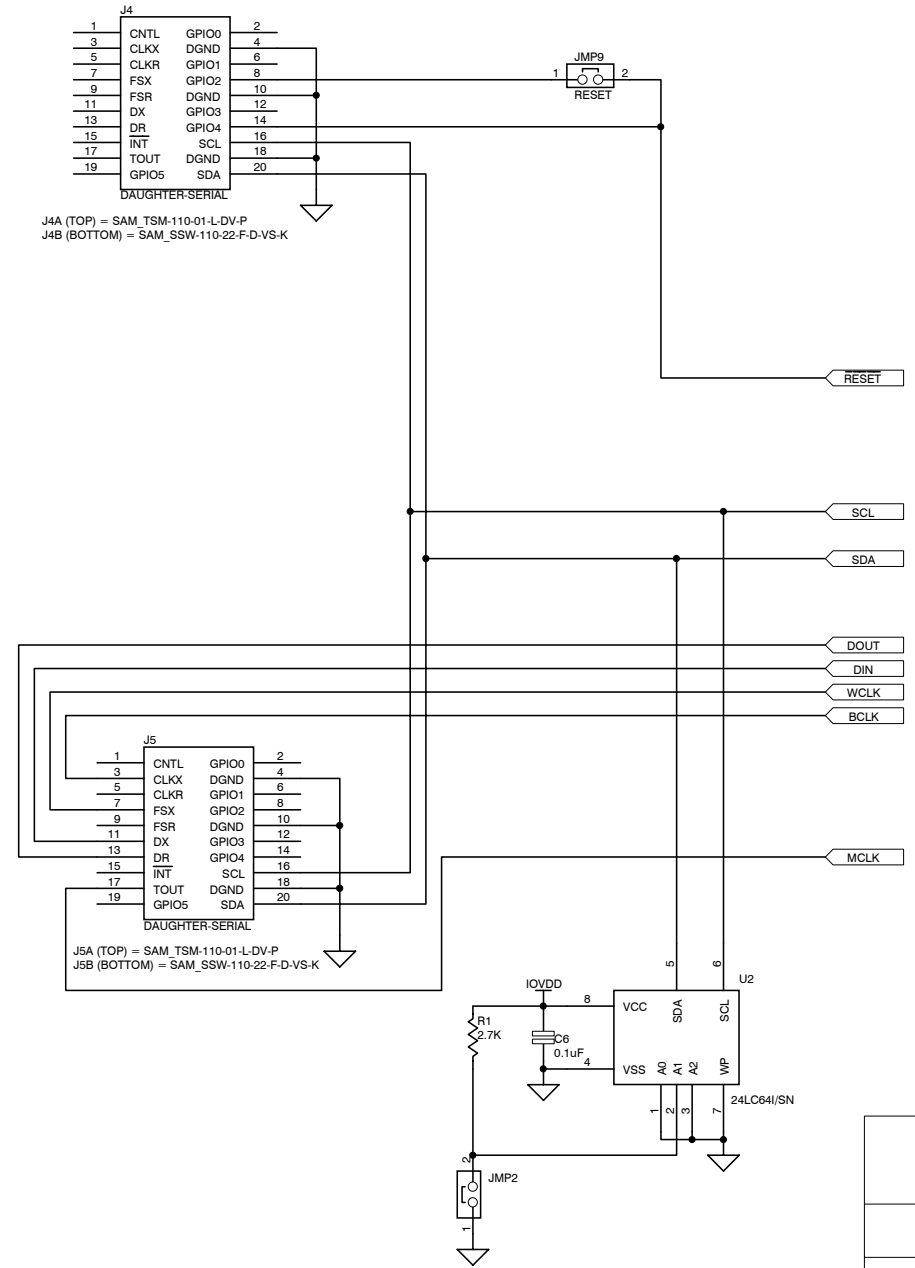
J4A (TOP) = SAM_TSM-110-01-L-DV-P
 J4B (BOTTOM) = SAM_SSW-110-22-F-D-VS-K



J2A (TOP) = SAM_TSM-110-01-L-DV-P
 J2B (BOTTOM) = SAM_SSW-110-22-F-D-VS-K



J3A (TOP) = SAM_TSM-105-01-L-DV-P
 J3B (BOTTOM) = SAM_SSW-105-22-F-D-VS-K



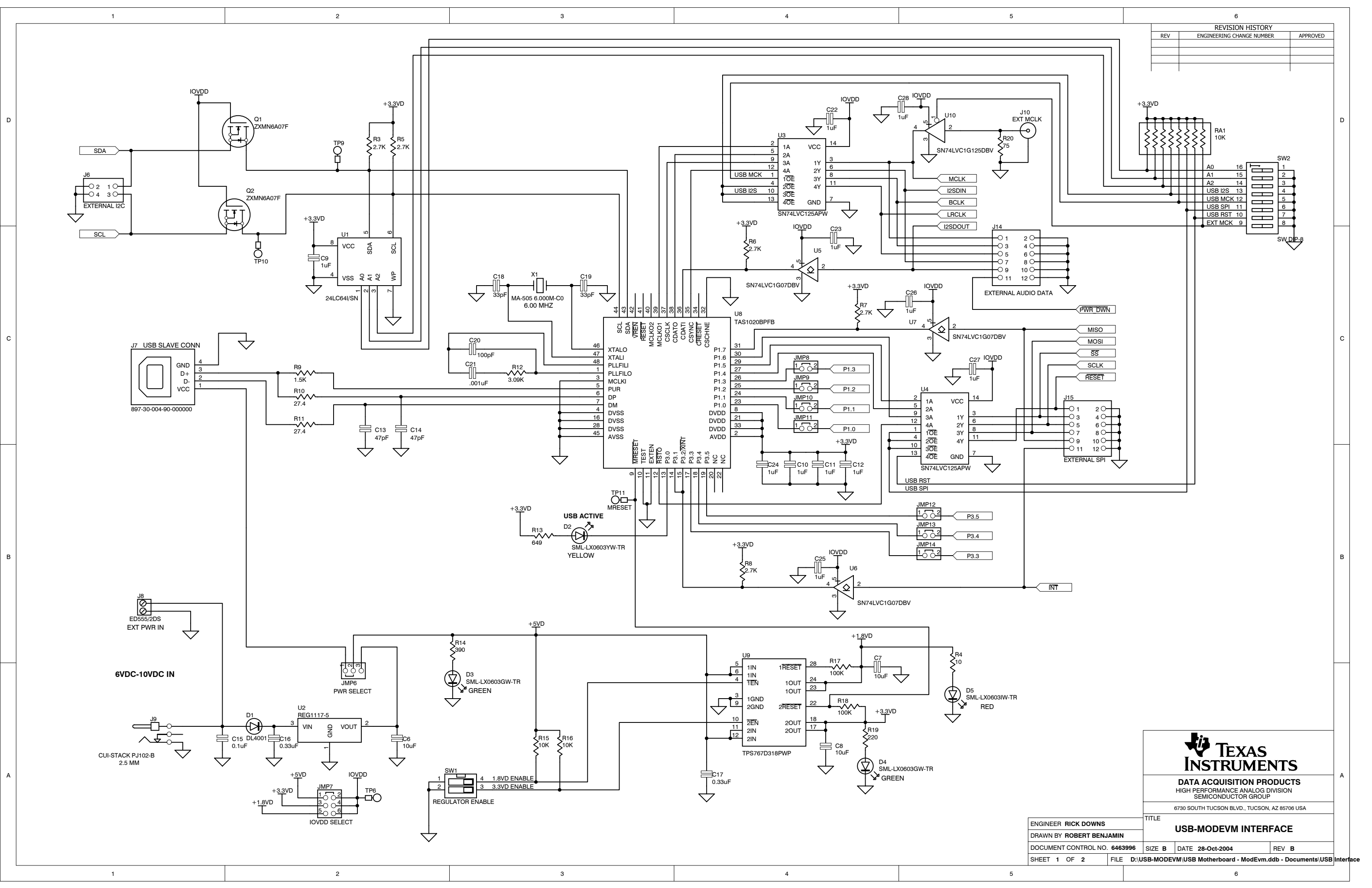
TEXAS INSTRUMENTS
 DATA ACQUISITION PRODUCTS
 HIGH-PERFORMANCE ANALOG DIVISION
 SEMICONDUCTOR GROUP
 6730 SOUTH TUCSON BLVD., TUCSON, AZ 85706 USA

ENGINEER	RICK DOWNS	TITLE	TLV320AIC32EVM INTERFACE
DRAWN BY	BOB BENJAMIN	DOCUMENT CONTROL NO.	6465231
SIZE	B	DATE	21-Mar-2005
SHEET	2 OF 2	REV	A
FILE	C:\Work\AIC32\TLV320AIC32EVM.Ddb - SCH\Daughtercard Interface		

Appendix B USB-MODEVM Schematic

The schematic diagram is provided as a reference.

REVISION HISTORY		
REV	ENGINEERING CHANGE NUMBER	APPROVED



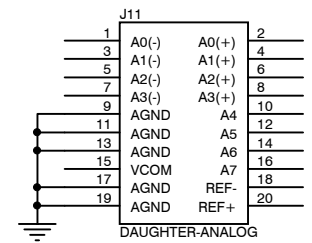
TEXAS INSTRUMENTS

DATA ACQUISITION PRODUCTS
HIGH PERFORMANCE ANALOG DIVISION
SEMICONDUCTOR GROUP

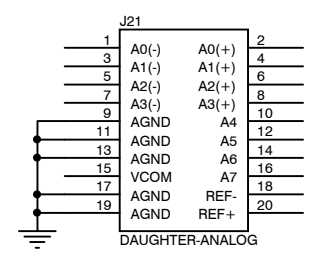
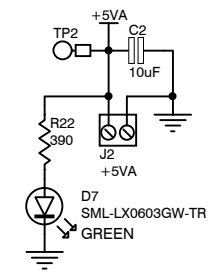
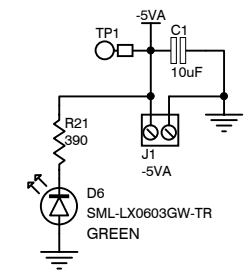
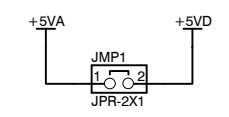
6730 SOUTH TUCSON BLVD., TUCSON, AZ 85706 USA

ENGINEER RICK DOWNS		TITLE	
DRAWN BY ROBERT BENJAMIN		USB-MODEVM INTERFACE	
DOCUMENT CONTROL NO. 6463996	SIZE B	DATE 28-Oct-2004	REV B
SHEET 1 OF 2	FILE D:\USB-MODEVM\USB Motherboard - ModEvm.ddb - Documents\USB Interface		

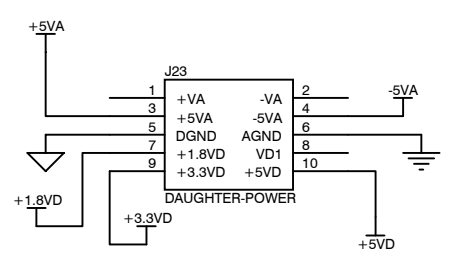
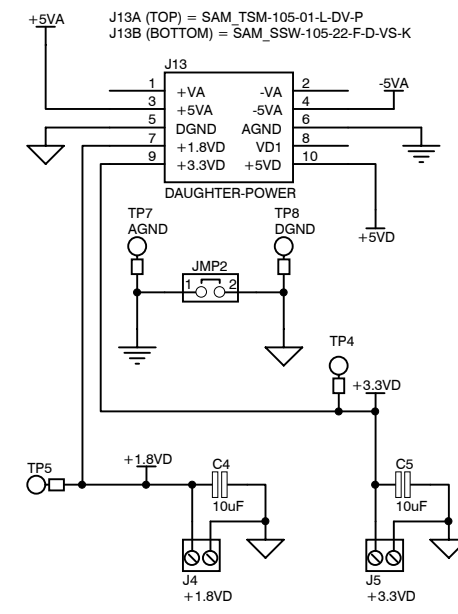
REVISION HISTORY		
REV	ENGINEERING CHANGE NUMBER	APPROVED



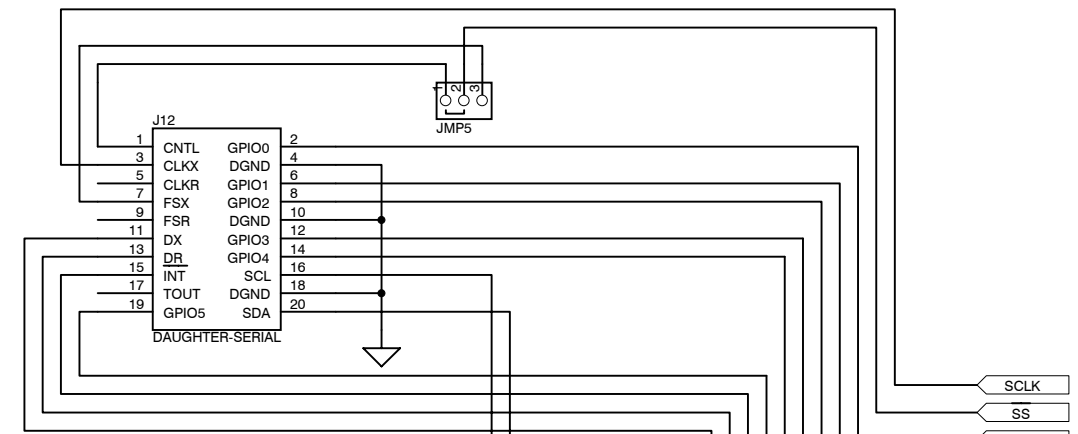
J11A (TOP) = SAM_TSM-110-01-L-DV-P
 J11B (BOTTOM) = SAM_SSW-110-22-F-D-VS-K



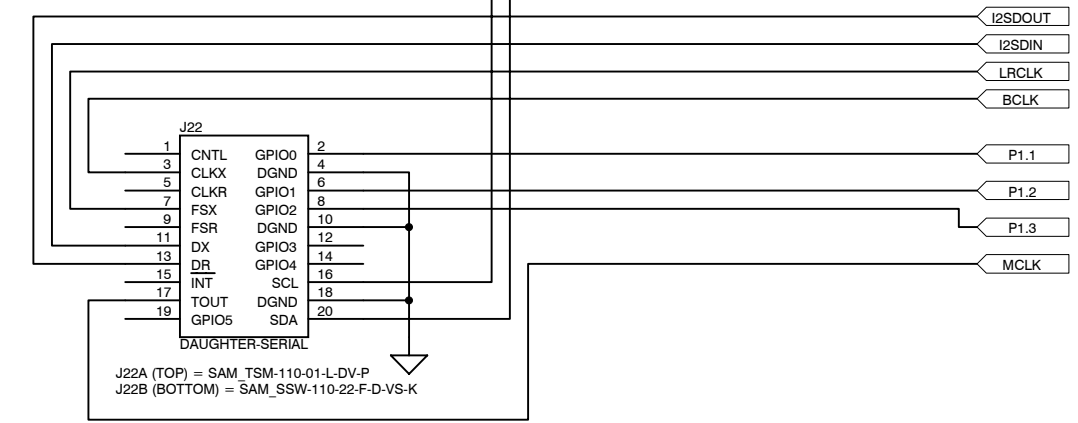
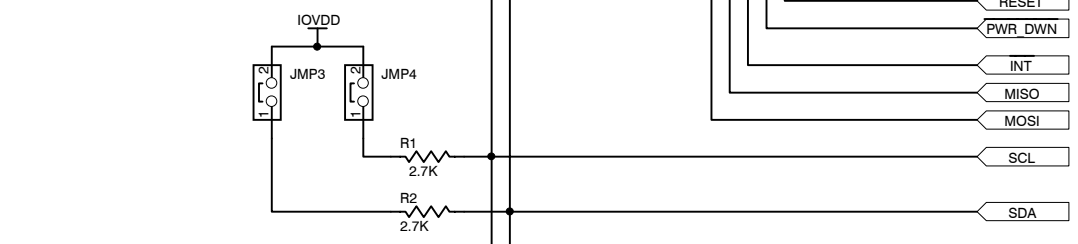
J21A (TOP) = SAM_TSM-110-01-L-DV-P
 J21B (BOTTOM) = SAM_SSW-110-22-F-D-VS-K



J23A (TOP) = SAM_TSM-105-01-L-DV-P
 J23B (BOTTOM) = SAM_SSW-105-22-F-D-VS-K



J12A (TOP) = SAM_TSM-110-01-L-DV-P
 J12B (BOTTOM) = SAM_SSW-110-22-F-D-VS-K



J22A (TOP) = SAM_TSM-110-01-L-DV-P
 J22B (BOTTOM) = SAM_SSW-110-22-F-D-VS-K

- SCLK
- SS
- P3.4
- P3.5
- P1.0
- RESET
- PWR DWN
- INT
- MISO
- MOSI
- SCL
- SDA
- I2SDOUT
- I2SDIN
- LRCLK
- BCLK
- P1.1
- P1.2
- P1.3
- MCLK



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	28-Oct-2004	REV	B	
SHEET	2 OF 2	FILE	D:\USB-MODEVM\USB Motherboard - ModEvm.ddb - Documents\Daughtercard Interface	

FCC Warnings

This equipment is intended for use in a laboratory test environment only. It generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to subpart J of part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment in other environments may cause interference with radio communications, in which case the user at his own expense will be required to take whatever measures may be required to correct this interference.

EVM TERMS AND CONDITIONS

Texas Instruments (TI) provides the enclosed Evaluation Module and related material (EVM) to you, the user, (you or user) **SUBJECT TO** the terms and conditions set forth below. By accepting and using the EVM, you are indicating that you have read, understand and agree to be bound by these terms and conditions. **IF YOU DO NOT AGREE TO BE BOUND BY THESE TERMS AND CONDITIONS, YOU MUST RETURN THE EVM AND NOT USE IT.**

This EVM is provided to you by TI and is intended for your **INTERNAL ENGINEERING DEVELOPMENT OR EVALUATION PURPOSES ONLY**. It is provided "AS IS" and "WITH ALL FAULTS." It is not considered by TI to be fit for commercial use. As such, the EVM may be incomplete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety measures typically found in the end product. As a prototype, the EVM does not fall within the scope of the European Union directive on electromagnetic compatibility and therefore may not meet the technical requirements of the directive.

Should this EVM not meet the specifications indicated in the EVM User's Guide, it may be returned within 30 days from the date of delivery for a full refund of any amount paid by user for the EVM, which user agrees shall be user's sole and exclusive remedy. **THE FOREGOING WARRANTY IS THE EXCLUSIVE WARRANTY MADE BY TI TO USER, AND IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING ANY WARRANTY OF MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE OR NON-INFRINGEMENT.**

TI shall have no obligation to defend any claim arising from the EVM, including but not limited to claims that the EVM infringes third party intellectual property. Further, TI shall have no liability to user for any costs, losses or damages resulting from any such claims. User shall indemnify and hold TI harmless against any damages, liabilities or costs resulting from any claim, suit or proceeding arising from user's handling or use of the EVM, including but not limited to, (i) claims that the EVM infringes a third party's intellectual property, and (ii) claims arising from the user's use or handling of the EVM. TI shall have no responsibility to defend any such claim, suit or proceeding.

User assumes all responsibility and liability for proper and safe handling and use of the EVM and the evaluation of the EVM. TI shall have no liability for any costs, losses or damages resulting from the use or handling of the EVM. User acknowledges that the EVM may not be regulatory compliant or agency certified (FCC, UL, CE, etc.). Due to the open construction of the EVM it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge.

EXCEPT TO THE EXTENT OF THE USER'S INDEMNITY OBLIGATIONS SET FORTH ABOVE, NEITHER PARTY SHALL BE LIABLE TO THE OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES WHETHER TI IS NOTIFIED OF THE POSSIBILITY OR NOT.

TI currently deals with a variety of customers for products, and therefore our arrangement with the user **is not exclusive**.

TI assumes **no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein.**

User agrees to read the EVM User's Guide and, specifically, the EVM warnings and Restrictions notice in the EVM User's Guide prior to handling the EVM and the product. This notice contains important safety information about temperatures and voltages.

It is user's responsibility to ensure that persons handling the EVM and the product have electronics training and observe good laboratory practice standards.

By providing user with this EVM, product and services, TI is **NOT** granting user any license in any patent or other intellectual property right.

EVM WARNINGS AND RESTRICTIONS

It is important to operate this EVM within the input voltage range of 3.3 V to 5 V and the output voltage range of 0 V to 5 V.

Exceeding the specified input range may cause unexpected operation and/or irreversible damage to the EVM. If there are questions concerning the input range, please contact a TI field representative prior to connecting the input power.

Applying loads outside of the specified output range may result in unintended operation and/or possible permanent damage to the EVM. Please consult the EVM User's Guide prior to connecting any load to the EVM output. If there is uncertainty as to the load specification, please contact a TI field representative.

During normal operation, some circuit components may have case temperatures greater than 30°C. The EVM is designed to operate properly with certain components above 85°C as long as the input and output ranges are maintained. These components include but are not limited to linear regulators, switching transistors, pass transistors, and current sense resistors. These types of devices can be identified using the EVM schematic located in the EVM User's Guide. When placing measurement probes near these devices during operation, please be aware that these devices may be very warm to the touch.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2005, Texas Instruments Incorporated

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 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.

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.

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
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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