Sample Code for Interfaceing the TLV320AIC1106 Codec With the TMS320C5402™ DSP

Bolanle Onodipe — *High Precision Analog/Data Acquisition Products*

ABSTRACT

The TLV320AIC1106 is one of Texas Instrument's (TI) high-performance codecs designed for easy use in various voice-band communication applications. This application report provides the user with a simple software package that facilitates the interface of this codec with the TMS320C5402™ DSP. It provides a good basis for more complex system developments. This report highlights some of the often-overlooked design issues that one should give consideration. The sample code discussed in this application report can be downloaded from http://www.ti.com/lit/zip/SLAA147.

Contents

List of Figures

List of Tables

1 Introduction

Texas Instruments offers a wide range of high performance codecs that can be used in various voice-band communication applications. The TLV320AIC1106 device is a low-power pulse code modulation (PCM) codec designed to perform analog-to-digital (A/D) conversion (coding), digital-to-analog (D/A) conversion (decoding), and transmit and receive filtering for voice-band systems that meet the Consultative Committee on International Telegraphy and Telephony (CCITT) g.714 requirements. It has a pin-selectable data-formatting mode that allows operation as a 13-bit linear device or an 8-bit companded (μ-Law) device. All internal clocks required by the device are generated automatically from the input 2.048-MHz master clock. There is also an integrated voice-band band pass filter whose passband ranges from 300 Hz to 3.4 kHz.

The AIC1106 is simple to interface because there are no registers to program. Only the receive volume control can be modified. This option is only available if the 13-bit linear mode is selected. TI has introduced an inexpensive AIC1106 evaluation module (EVM), the TLV320AIC1106EVM, that is easily interfaced with a C54x™ DSP development platform [reference 2].

Figure 1 shows the TLV320AIC1106EVM.

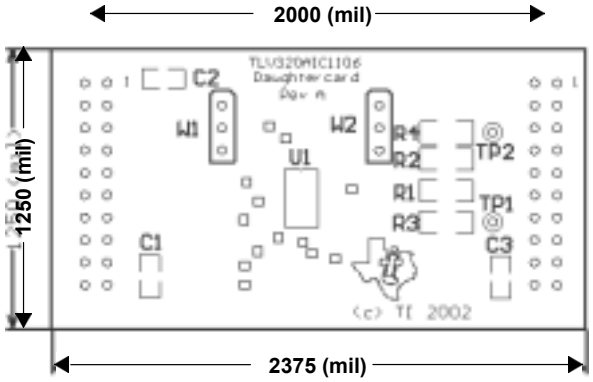

Figure 1. TLV320AIC1106 Codec EVM

The C5402™ DSP is a fixed-point DSP with 16-K word on-chip memory and various on-chip peripherals. The C5402™ DSP performs up to 100 million instructions per second (MIPS), while the C5402A™ DSP performs up to 160 MIPS. These peripherals include two multichannel buffered serial ports (McBSP's), an enhanced 8-bit parallel host port, two 16-bit timers, and a six-channel direct memory access (DMA) controller [reference 6]. TI also has a series of DSP starter kits (DSK) for the various DSP families, in this case the C5402™ DSK. When interfaced with this starter kit, the DSP systems developer can design and test a complete AIC1106 codec system.

This application report describes the software needed to interface the AIC1106 codec with the C54xx™ DSP. It also provides a tested and proven software example code.

For more information on the C5402™ DSP see reference 7, and for more information on the AIC1106 codec, see references 2 and 5.

2 Design Concept

The AIC1106 codec is a voice-band communication device that interfaces easily C5402™ DSK through the DSK's McBSP port. See the Section 2.4 TLV320AIC1106 Control and Section 2.7 on the McBSP initialization for detailed information on configuring both the AIC1106 codec and the McBSP.

The codec operates as a stand-alone slave that receives all its timing and synchronization signals from the DSP, the master. The DSP is programmed to generate a 2.048-MHz clock signal, and an 8-kHz frame synchronization signal. These signals are made available at the CLKX and FSX terminals of McBSP1 respectively. They both drive the MCLK and the PCMSYNC inputs of the codec. The analog-to-digital conversion output of the codec is available at the PCMO terminal of the codec, and the digital input to the digital-to-analog converter is fed into the codec through the PCMI terminal.

This device is a 13-bit codec with a configurable data I/O format. The data format is a 13-bit, left justified, linear data format, or an 8-bit, left justified, companded μ-law data format.

The basic concept of this example is as follows:

- Sample a given analog input signal at 8 kHz through the receive terminal of the DSP.
- Append each sampled 13-bit data with a 3-bit receiver volume gain control code, and reroute the results to the digital-to-analog converter of the codec through the DSP's transmit terminal.

3 Software Interface

For the interaction between the DSP and the codec, a set of software routines was developed. These include the DSP initialization routine for both the McBSP and the complex programmable logic device (CPLD) registers and the interrupt service routines for the DSP/codec handshake. This software package is available for download, see the AIC1106.zip file in the product folder. The user can adapt this software package for whatever application they wish to implement. In addition, shell programs or templates for customizing application software programs are also available [see reference 4].

3.1 DSP Initialization

The DSP initialization routine sets up the DSP by programming its memory-mapped registers (MMRs) to the top of the data-memory page. For a more comprehensive discussion of the MMRs, [see reference 6]. The MMRs are defined in the MMRegs.h file in the software package down load file, see the AIC1106.zip file in the product folder.

3.2 DSP System Clock Frequency

An onboard 20-MHz crystal oscillator provides the basic system clock for the C5402™ DSK. The on-chip phase locked loop (PLL) circuitry provides a series of derivative clock frequencies that can be selected by meeting either of the following conditions:

- Hardware, flipping the posts of the DIP switches DIP SW #5, DIP SW #4 and DIP SW#3 that correspond to the terminals CLKMD1, CLKMD2 and CLKMD3 respectively. See the CLKMD DIP switch control chart table in Table 1. The setting programmed through this DIP switch is the reset default frequency of the DSP.
- Software, setting the corresponding bits of MMR CLKM changes the DSP clock frequency from the reset default value to a new operating value. Refer to *Interfacing the TLV320AIC10/11 to the TMS320C5402 DSP*, SLAA109 [reference 1], for more information.

Table 1. DIP Switch Settings for CLKMD Configuration

DIP SW #5 (CLKMD1)	DIP SW #4 (CLKMD2)	DIP SW #3 (CLKMD3)	CLKMD RESET DEFAULT	DSP CPU CLOCK FREQUENCY (DSK ONBOARD 20–MHz CRYSTAL)
0	0	0	0xE007	x15 (not valid)
0	0	1	0x9007	1 x10 (not valid)
0	1	0	0x4007	x5 (100 MHz)
1	0	0	0x1007	x2 (40 MHz)
1	1	0	0xF007	x1 (20 MHz)
1	1	1	0x0000	x0.5 (10 MHz)
1	0	1	0xF000	x0.25 (5 MHz)
0	1	1	—	Reserved

3.3 Software Loop Control

The main program for this software release has an infinite loop from within which a loop control routine is called continuously. It allows the loop to be repeated at a fixed frequency of 16 kHz. Its operation and cycling frequency are described in the subsequent paragraphs. If the user wishes to include some routines that should be repeated periodically, the routines should be placed within this infinite loop. These routines should be short enough to be completed within the 0.0625 msec period; otherwise the 16-kHz period requirement must be relaxed.

Timer0 sets and controls an idle loop that determines the rate at which the instructions in the main loop of the main program are repeated. For this example, a cycling rate of 16 kHz is used. Assume that the DSP is configured for a 100-MHz clock rate, the required loop rate can be obtained by meeting any of the following conditions:

- Setting the Timer0 divide-down counter, MMR:TDDR to 0 and MMR:PDR to 6249, on the basis of the LoopFrequency = 100 MHz / (MMR:TDDR + 1) / (MMR:PDR + 1) formula.
- Setting the Timer0 divide-down counter, MMR:TDDR to 4 and MMR:PDR to 1249, on the basis of the LoopFrequency = 100 MHz / (MMR:TDDR + 1) / (MMR:PDR + 1) formula.
- Setting any combination of MMR:TDDR and MMR:PDR that would make the product of (MMR:TDDR + 1) and (MMR:PDR + 1) equal to 6250.

Upon executing the InitC5402 () routine, the Timer0 is stopped, the timer's MMR registers are programmed for the 16 kHz requirement. The Timer's count down is subsequently kicked off, and the infinite loop of the main program is entered. After executing the body of the main program, the LoopControl () routine continuously monitors the status of the Timer0's interrupt flag. Note that the timer's interrupt does not have to be enabled since its status is being polled. Upon time-out, the interrupt flag is cleared, and control is passed back to the main loop, where the body of the main loop is repeated. This cycle continues indefinitely. Refer to the flow chart depicted in Figure 2.

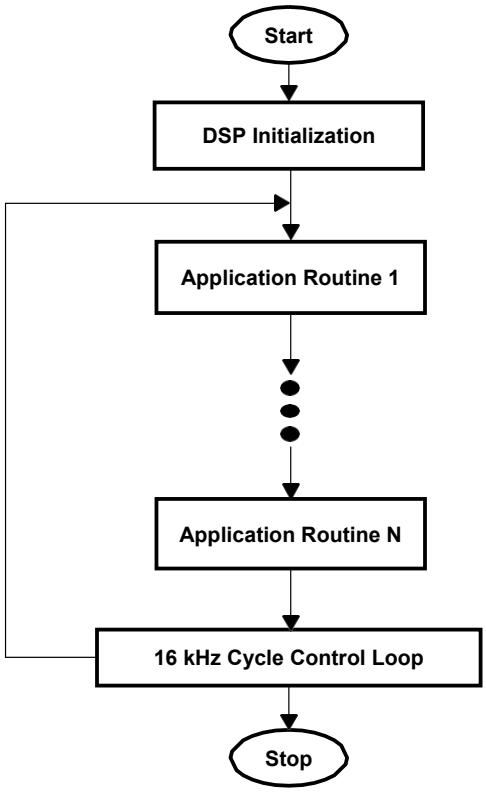

Figure 2. Main Loop Control Flow Chart

3.4 McBSP Initialization

The McBSP is a high-speed, full-duplex serial port communication channel. There are two independent McBSP channels on this DSP, the McBSP0 and the McBSP1. The McBSP's triple buffered input port (receiver) and double buffered output port (transmitter) on each channel enables a continuous stream of input and output data to be transferred across either channel. In addition, an independent setting for each channel's frame synchronization and clocking signals for data transmitting and data receiving can be selected. When properly configured, a McBSP channel interfaces gluelessly with another McBSP channel, any individual member of the AIC family, or any other device that adheres to the smart time division multiplexed serial port (SMARTDM) communication protocol. The ability to configure these clock signal terminals, frame synchronization and clocking, as inputs or outputs, gives the McBSP the flexibility to operate as a master device or a slave device.

For this application report, the DSP is the master, and it communicates with the slave TLV320AIC1106 codec device, through the McBSP1.

See Table 2 for the pin definition and pin description for each McBSP channel.

Table 2. McBSP Hardware Pin Description

PIN ID	I/O DIRECTION	DESCRIPTION
DR	Input	Received serial data
DX	Input / Output / High–Impedance	Transmitted serial data
FSR	Input / Output / High–Impedance	Receive frame synchronization
FSX	Input / Output / High–Impedance	Transmit frame synchronization
CLKR	Input / Output / High–Impedance	Receive clock
CLKX	Input / Output / High–Impedance	Transmit clock
CLCKS	Input	External McBSP System Clock

Each McBSP channel has seven pins, through which the DSP communicates with the outside world. The direction and characteristics of each terminal are programmed through the McBSP's MMR registers described in Table 3. Refer to *TMS320C54x DSP Enhanced Peripherals, Reference Set Volume 5, SPRU302* [see reference 8], for more information on accessing the McBSP control registers.

Table 3. McBSP Registers Memory Map

ADDRESS DATA MEMORY		SUBADDRESS	ACRONYM	DESCRIPTION
McBSP0	McBSP1			
0x0020	0x0040		DRR2	Data receive register 2
0x0021	0x0041		DRR1	Data receive register 1
0x0022	0x0042		DXR2	Data transmit register 2
0x0023	0x0043		DXR1	Data transmit register 1
0x0038	0x0048		SPSA	Serial port subbank address register
0x0039	0x0049		SPSD	Serial port subbank data register
		0x0000	SPSCR1	Serial port control register 1
		0x0001	SPSCR2	Serial port control register 2
		0x0002	RCR1	Receive control register 1
		0x0003	RCR2	Receive control register 2
		0x0004	XCR1	Transmit control register 1
		0x0005	XCR2	Transmit control register 2
		0x0006	SRGR1	Sample rate generator register 1
		0x0007	SRGR2	Sample rate generator register 2
		0x0008	MCR1	Multichannel register 1
		0x0009	MCR2	Multichannel register 2
		0x000A	RCERA	Receive-channel enable register partition A
		0x000B	RCERB	Receive-channel enable register partition B
		0x000C	XCERA	Transmit-channel enable register partition A
		0x000D	XCERB	Transmit-channel enable register partition B
		0x000E	PCR	McBSP pin-control register

For this application report, the McBSP1 is programmed such that

- The word size is 16-bit.
- Data is left-justified.
- Frame length is 1 word per frame.
- Transmit interrupts are issued on new-frame synchs.
- The frame synch rate, i.e., sampling frequency is 8 kHz.
- CLKX or MCLK is set for 2.048 MHz.

3.5 Interrupts

There are two types of interrupt systems on this DSP, the hardware driven interrupts which are truly synchronous, and the software driven interrupts which are asynchronous. The C5402™ DSP has 30 different interrupt vectors.

- Two nonmaskable interrupts, the hardware RESET and the nonmaskable interrupt (NMI). These interrupt sources can not be disabled by software. The global interrupt enable (GIE) status has no bearing on the individual nonmaskable interrupt's ability to interrupt any execution process.
- Fourteen software interrupts that can be issued at any point during the program execution. If called within a subroutine, as long as the DSP's interrupt system is globally enabled, and there are no interrupts of higher priority, the associated interrupt service routine (ISR) is called.
- Fourteen hardware peripheral interrupts that could be enabled or disable corporately by enabling or disabling the DSP's interrupt system globally, or individually, at the designers pleasure. If all the following conditions are satisfied:
  - The DSP's interrupt system is globally enabled.
  - That particular interrupt is individually enabled.
  - There is no other interrupt with a higher priority being serviced.

  At the end of the current instruction, the CPU suspends the current routine being processed, and vectors to the associated interrupt service routine (ISR).

Upon receiving a valid interrupt request that satisfies the requirements stated above, the DSP accesses an interrupt vector table from which it picks up the address of the corresponding ISR routine. The application report *Interfacing the TLV320AIC10/11 to the TMS320C5402 DSP*, SLAA109 [reference 1], provides a more concise description of the C5402™ DSP interrupt protocol and ISR vectoring.

The McBSP1 is configured to generate a transmit interrupt (XINT) each time a frame synchronization pulse is issued. When this occurs, with no interrupts of higher priority pending, and the McBSP1's XINT interrupt enabled, then the DSP suspends its current tasks after completing the current instruction and vectors to the appropriate ISR routine, McBSP1TXISR(). The ISR assumes that upon issuing a frame-sync pulse, a string of 16-bit digital data, the result of a previous conversion, is available for reception, and that the codec is also ready to receive a 16-bit string of digital data from the DSP. Within this routine, the DSP reads the result of the codec's left-justified analog-to-digital conversion through the McBSP1's DRR1 register, appends the result with the desired receive volume gain, and sends the new data out to the codec's digital-to-analog converter through the McBSP1's DXR1 register. This process simulates a digital loopback system.

The ISR used for this application example is available in the software package down load file, see the AIC1106.zip file in the product folder.

3.6 CPLD Register Initialization

The C5402 DSK has a complex programmable logic device (CPLD) through which it implements the various logic required for interfacing the DSP to the outside world, and at the same time, providing control and status interfaces for the DSP through software. Refer to *Interfacing the TLV320AIC10/11 to the TMS320C5402 DSP*, SLAA109 [see reference 1], for more details.

The C5402 DSK is equipped with an onboard codec unit. The onboard codec is configured to be interfaced with the DSP through the McBSP port 0, and the onboard data access arrangement (DAA) chip is also interfaced through the McBSP port 1. However, there are only two McBSP ports on this DSP. In order to accommodate a plug-in codec board, there must be a way to reroute the McBSP ports so they are alternatively connected through software to an external unit. This is one of the functions of the CPLD. By programming various bit patterns into the control register CNTL2 of the CPLD, the user chooses between routing the McBSP signals to the various onboard peripherals, or to the plugged-in peripherals (see Figure 3). Refer to Table 4 for the definition of the bits of the CLPD control register CNTL2. The appended AIC initialization assembly routine includes the set of commands required to accomplish this signal rerouting.

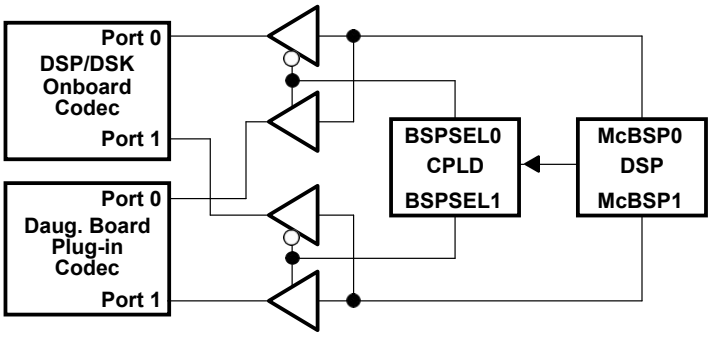

Figure 3. CLDP Control Schematic

Table 4. CLDP Control Register, CNTL2

BIT POSITION	ACRONYM	R/W	DESCRIPTION
7	DAAOH	RW	DAA off-hook control
6	DAACID	RW	DAA caller ID enable
5	FLASHENB	RW	External memory source selection (0 = Flash; 1 = SRAM)
4	INT1SEL	RW	INT1 interrupt source selection (0 = UART; 1 = daughter board)
3	FC1CON	RW	Mic/speaker AD50 FC control
2	FC0CON	RW	DAA AD50 FC control
1	BSPSEL1	RW	McBSP1 data source selection (0 = Mic/speaker; 1 = daughter board)
0	BSPSEL0	RW	McBSP0 data source selection (0 = DAA; 1 = daughter board)

3.7 TLV320AIC1106 Control

The AIC1106 has the capability of selecting between a linear data format and a logarithmic data format. If a logic 0 is applied to the $\overline{\text{LINSEL}}$ terminal of the AIC1106, the device is placed in a 13-bit left-justified linear mode. If a logic 1 is applied to the $\overline{\text{LINSEL}}$ terminal of the AIC1106, the device is placed in a logarithmic, 8-bit left-justified μ-Law companded mode (see Table 5). In addition, the AIC1106 codec has two mute control terminals, one for muting the microphone, and the other for muting the earphone. Jumper settings on the TLV320AIC1106EVM required to control the codec's mute facility are given in Table 6.

Table 5. Data Conversion Format

$\overline{\text{LINSEL}}$ LOGIC STATE	DATA FORMAT MODE
0	Left-justified 13-bit linear conversion
1	Left-justified 8-bit μ-Law companded conversion

Table 6. Functions and Default Settings of Jumpers W1 and W2 for the TLV320AIC1106EVM

JUMPER	DESCRIPTION	POSITION		
		1–2	2–3	DEFAULT
W1	Microphone mute selection	Mute	No mute	No mute
W2	Earphone mute selection	Mute	No mute	No mute

Refer to the application report [see reference 2] on the AIC1106 hardware, for more information on the hardware settings for this device.

In the 13-bit linear mode, the three least significant bits slot of the available 16-bit data stream that is transmitted to the codec is used to convey the receiver volume gain code to the codec through the PCMI terminal. From this 16-bit string, the 13-bit digital DAC information is extracted and passed on to the DAC, and the three LSBs are passed on to the volume control circuitry, forming the receiver volume gain code (RXVOL [2:0]).

Note that if the companding option was selected, i.e., an 8-bit μ-Law format; then the ability to program the receiver volume gain is negated, and the default gain is fixed at 0 dB.

The desired gain is programmed into the least significant 3-bit slot of the 16-bit input PCMI data stream (see Table 7). In the companding 8-bit mode, the volume gain is fixed at 0 dB.

Table 7. Volume Control Bit Definition in Linear Mode

RXVOL [2:0]	GAIN SETTING
000	3 dB
001	0 dB
010	–3 dB
011	–6 dB
100	–9 dB
101	–12 dB
110	–15 dB
111	–18 dB

3.8 Codec's ADC/DAC Sampling Frequency

A 2.048-MHz master clock (MCLK) signal is delivered from the CLKX terminal of McBSP1 to the codec's MCLK terminal. In addition, the frame synchronization signal, which is also the sampling frequency signal, assumes values up to 8 kHz. It is transmitted from the FSX terminal of McBSP1 to the PCMSYNC terminal of the codec. From both input signals, all the derivative clocks required by the codec are generated internally.

3.9 Design Issues

The AIC1106 codec is easy to use because it does not have a configurable register. However, the following items must be considered:

- The $\overline{\text{LINSEL}}$ must be at the proper logic.
- The AIC1106 device has an integral PLL set at 2.048 MHz. The MCLK should be very close to that value.
- The conversion input and output data stream are left justified and are zero-filled in either case, linear 13-bit or 8-bit μ-law companding. Therefore, if the user wishes to apply a gain to the DAC codec, there is no reason to shift a previously converted data stream left by three bits before adding the 3-bit value of RXVOL [2:0]. For instance, if a –12 dB attenuation is applied to an A/D codec result, 101b must be added to the previous A/D result, and transmitted to the D/A codec. This was implemented in the appended software package (see Figure 4).

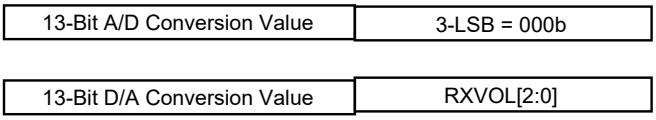

Figure 4. 13-Bit Linear Mode A/D and D/A Data Format

4 References

1. *Interfacing the TLV320AIC10/11 to the TMS320C5402 DSP*, literature number SLAA109.
2. *Codec Evaluation System* (SLAA141)
3. Application report *TLV320AIC12/13/14/15 Codec Operating Under Stand-Alone Slave Mode* literature number SLAA142
4. *TMS320C54x DSP Reference Set, Volume 4: Applications Guid*e, Texas Instruments Literature number SPRU173
5. *TLV320AIC1106 PCM CODEC* data sheet, literature number SLAS357
6. *TMS320C54x DSP Reference Set*, Volume 1: *CPU and Peripherals*, Texas Instruments literature number SPRU131F
7. *TMS320VC5402 Fixed-Point Digital Signal Processor*, data sheet, Texas Instruments literature number SPRS079D
8. *TMS320C54x DSP Enhanced Peripherals, Reference Set* Volume 5, literature number SPRU302

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

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