TLV320AIC12/13/14/15 Codec Operating In Stand-Alone Slave Mode

Wendy X. Fang, PhD. *High Performance Analog Group*

ABSTRACT

This application report presents the methods, tips, and examples for using the TLV320AIC12/13/14/15 codec in stand-alone slave mode. The TMS320C5402 DSK and the Code Composer Studio™ (CCS) were used as the development platform for this DSP-codec system. Project collateral discussed in this application report can be downloaded from the following URL: http://www.ti.com/lit/zip/SLAA142.

Contents

List of Figures

List of Tables

1 Introduction

An analog interface circuit (AIC), often called a codec (coder/decoder) device, is an electronic modem that contains an analog to digital converter (ADC or *coder*), and a digital to analog converter (DAC or *decoder*), on a single chip. In additional to the analog input/output ports, there is also one or more serial digital ports on an AIC device. These ports are the *bridges* between the AIC and a digital signal processor (DSP), a microcontroller unit (MCU), or other digital devices.

TI's TLV320AIC family of codec devices provides customers with a wide range of choices for the voice/audio band ADC and DAC applications. Each of the members within the TLV320AIC family offers design flexibility for their specific application. To apply these devices and make good use of their features, it is necessary to properly configure the device. This is accomplished by hardware and software (or firmware). This application report mainly addresses the application, configuration, and potential interface problems.

The four devices addressed in this application report are the TLV320AIC12, TLV320AIC13, TLV320AIC14, and TLV320AIC15. These four types of codec devices share the same core, and their differences are only in the voltage range of the digital input/output power supply, and the number of the analog output ports. The differences are summarized in Table 1. For more details on these devices, refer to the corresponding data sheets [1], [2], [3], and [4] listed in the reference section.

Table 1. Summary of AIC12/13/14/15 Differences

Codec	TLV320AIC12	TLV320AIC13	TLV320AIC14	TLV320AIC15
Power Supply for Digital I/O	2.7–3.6VDC	1.1–3.6VDC	2.7–3.6VDC	1.1–3.6VDC
Analog Outputs	OUTP1/OUTM1 and OUTP2, OUTP3 and OUTM	OUTP1/OUTM1 and OUTP2, OUTP3 and OUTM	OUTP1/OUTM1 Only	OUTP1/OUTM1 Only

The following discussion is applicable to each of these devices, unless otherwise indicated. For simplification, only the TLV320AIC12, or 'AIC12, is discussed throughout the remainder of this application report.

TI's TMS320C5402 DSK, and the associated Code Composer Studio (CCS), are utilized as the development platform. The discussion and firmware code are applicable for systems that use the 'AIC12 and C54xx interface.

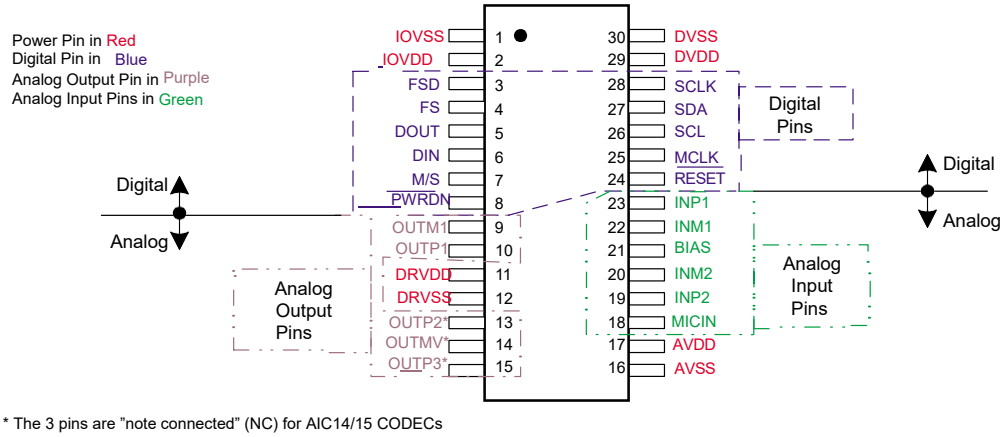
2 Hardware System

This section gives a fundamental and brief description of the DSP/codec system hardware.

2.1 TLV320AIC12 Stand-Alone Operation

The 'AIC12 codec is a low-cost, low-power, high-performance voice-band codec, featuring 16-bit resolution, at speeds up to 26 kilo-samples per second (ksps). There are two serial digital ports on the codec: one called the *data port* and another called the *host port*. The 4-wire data port is designed for directly interfacing with the multichannel buffered serial port (McBSP) found in many of TI's DSPs. Examples are the TMS32C28xx, TMS32C5xxx, and TMS32C6xxx. The two-wire host port supports I^2C or S^2C (start-stop communication) interface. The data port can be used for both voice data receiving/transmitting (RX/TX) and codec configuration, while the host port can only be used for codec configuration.

A closer look at the ‘AIC12 pinout in reference [1], page 2–1, shows the hardware pins classified into four groups, based on their function: (1) the power pins, (2) the digital pins, (3) the analog input pins, and (4) the analog output pins. Figure 1 shows the four groups highlighted with four different colors, and also groups the pins according to type. There are also four tables in the figure that correspond to the four signal groups. Some of the important application features are listed in these tables.



* The 3 pins are "note connected" (NC) for AIC14/15 CODECs

Power Supply Pins:

#	Name	Analog	Digital
2 1	IOVDD IOVSS		3.3VDC[1] DGND
29 30	DVDD DVSS		1.8VDC DGND
11 12	DRVDD DRVSS	3.3VDC AGND	
29 30	DVDD DVSS	3.3VDC AGND	

[1] IOVDDrange 2.7 ~ 3.6VDC at AIC12/14, and 1.1 ~ 3.6VDC at AIC13/15.

Digital Pins

#	Name	Input	Output	Note
6 5 28 4	DIN DOUT SCLK FS	✓ [2] [2]	✓	4–wire Serial data port
26 27	SCL SDA	✓ [3]	 [3]	2–wire Host port
3 7 8 24	FDS M/S PWRDN RESET	 ✓ ✓ ✓	✓	
25	MCLK	✓		Master Clock Up to 100 MHz

[2] SCLK & FS are inputs at the CODEC’s slave mode, but outputs at master mode.

[3] SDA is bi–directional.

Analog Output Pins

#	Name	Load	Note
10 9	OUTP1 OUTM1	>= 600 Ohms	Enable at default
13 14 15	OUTP2 OUTMV OUTP3	>= 16/32 Ohms	Enable by Software

[4] OUTP2, OUTP3 and OUTMV are available at AIC12/13 only.

Analog Input Pins

#	Name	Input Type	Note
23 22	INP1 INM1		Enable at default
19 20	INP2 INM2		Enable by Software
18 21	MICIN BIAS	Microphone	Enable by Software

Figure 1. ’AIC12/13/14/15 Hardware Pinout

The ‘AIC12 can be operated in a *stand-alone master* mode, *stand-alone slave* mode, or with two or more ‘AIC12s in *cascade* mode. The maximum number of ‘AIC12 devices that can be cascaded is 16. This report focuses on *stand-alone slave* mode operation only. Additional applications are being developed using the other operating modes of the codec. Refer to the TI website periodically for new application reports for the ‘AIC12.

To apply the ‘AIC12 codec in any application, a specific circuit is essential for supporting the proper operation of the device. Figure 2 shows a basic circuit around the codec in its stand-alone slave operation.

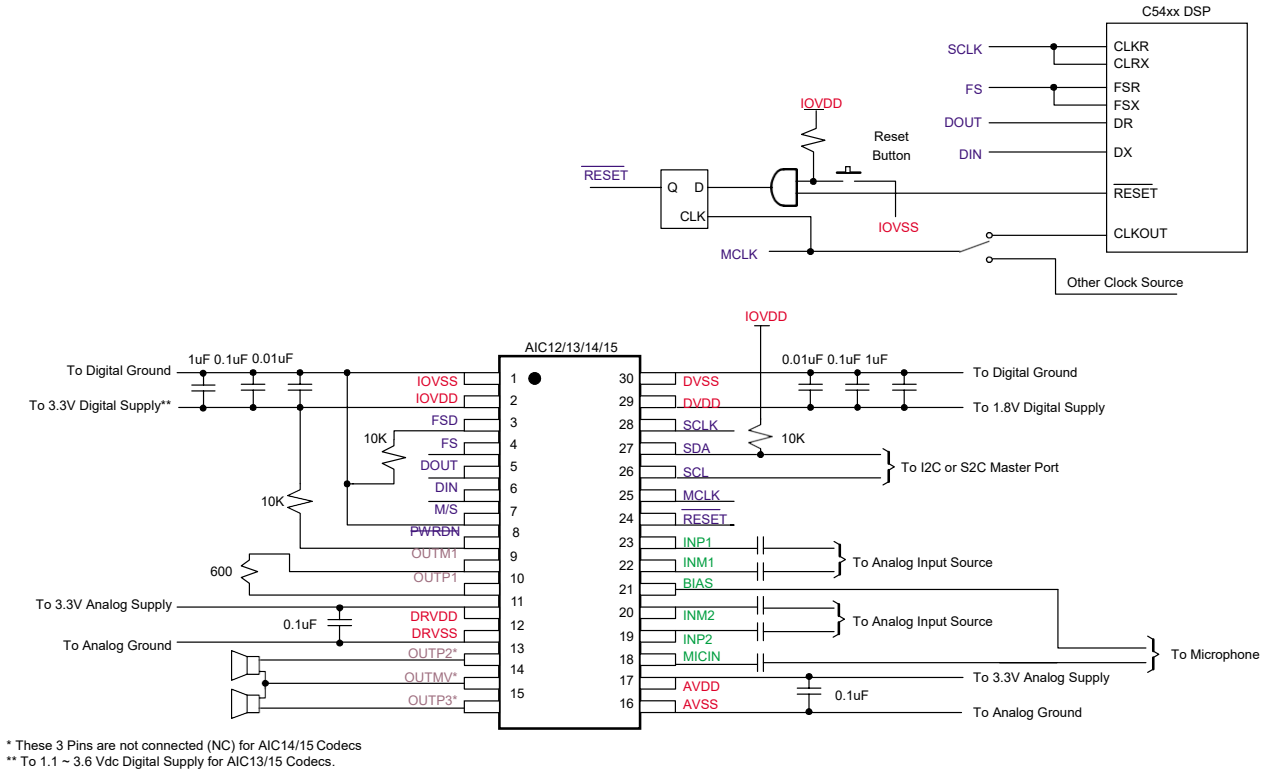


Figure 2. Basic Circuit for 'AIC12/13/14/15 in Stand-Alone Slave Mode

The M/S and FSD hardware pins of the 'AIC12 codec determine the mode of operation: as a stand-alone master, as a stand-alone slave, as a cascade master, or as a cascade slave. Table 2 lists all the 'AIC12 operation modes, and the relationship between the AIC mode and the logic status of the M/S and FSD pins.

Table 2. AIC12 Operating Mode Hardware Settings

AIC12/13/14/15		M/S (Pin#7)	FSD (Pin#3)
Operating Mode	Stand-Alone Master	High	Pull High
	Stand-Alone Slave	Low	Pull Low
	Cascade, Master	High	To Next Codec's FS pin
	Cascade, Slave in Middle Position	Low	To Next Codec's FS pin
	Cascade, Last Slave	Low	Pull High

2.2 TMS320C5402 DSK

The DSP/codec system information presented in this application report was developed using the TMS320C5402 DSK as the hardware platform and the 'C5402 DSK board as the mother board. The DSK board is included in the DSK package. References [5] through [7] provide detailed information on the DSP. For up-to-date information on the 'C5402 DSK, check TI's web site: *http://focus.ti.com/docs/tool/toolfolder.jhtml?PartNumber=TMDS320005402*

The 'C5402 DSK board has an 8-position DIP switch and four hardware jumpers/straps. The EVM operating mode and working condition are controlled by these parts, and require setup before connecting and powering-up the board. These settings are referred to as the *hardware configuration* in this report.

The hardware configuration of the ‘C5402 DSK board is set according to Tables 3 and 4.

Table 3. Configuration for DSK Board DIP Switch

SWITCH	NAME	ON/OFF	Description
1	JTAGSEL	OFF	Use external (e.g., XDS510PP Emulator)
2	MP/MC	ON	At microcontroller mode
3	CLKMD3	ON	The CLKMD pins are set to 0 1 0, That is: default DSP CPU clock frequency = 100MHz (20 MHz X 5).
4	CLKMD2	OFF	
5	CLKMD1	ON	
6	DMSEL	OFF	External memory is onboard
7	USER 1	ON	User S/W defined (0)
8	USER 0	ON	User S/W defined (0)

Table 4. Configuration for DSK Board Straps

STRAP NAME	SETTING	DESCRIPTION
JP1	1 to 2	CPLD program via J1
JP2	2 to 3	Normal boot mode
JP3	2 to 3	Low-impedance output for speaker driver
JP4	Installed	125 mA DAA loop current

NOTE: All strap/jumper settings listed are the default settings by the manufacturer.

As mentioned above, the ‘C5402 DSK board was used as the EVM system motherboard for developing this report. The *TLV320AIC / DSP development* board (or daughter board), was connected through the 80-pin *enhanced peripheral interface* (EPI) connector on the ‘C5402 DSK board. Refer to [8] for the actual definition of the EPI. The EPI can also be found on many other DSP EVM boards from TI, for example: the ‘C67xx DSK.

The following ‘C5402 DSP signals, conveyed through the EPI connector, may be used with the ‘AIC12 codecs:

- DX, DR, FSX, FSR, CLKX, CLKR, and CLKS are signals for McBSP0 or McBSP1 ports. They are used for voice data RX/TX from/to the ‘AIC12 and for the codec software configuration.
- CLKOUT (half the CPU clock) is the clock from the DSP, and it can be used for the codec main clock MCLK.
- $\overline{\text{RESET}}$ is the reset signal from the DSK board.
- ±12 VDC, 5 VDC, and 3.3 VDC from the DSK power supplies are also furnished through the EPI connector. The 3.3V DC power supply is regulated and has a maximum power capacity of 1 ampere.

2.3 TLV320AIC Codec EVM

The TI TLV320AIC family codec evaluation module (EVM) system was used for developing this report. Refer to [9] for more details on the AIC EVM system. A *TLV320AIC / DSP development* board and an ‘AIC12 *EVM* board were used for the stand-alone slave codec operation.

The development board can be used to evaluate and test most of the devices within the TLV320AIC family. An 80-pin connector on the board interfaces with the DSK. Another connector brings signals from the serial ports to the corresponding AIC EVM board(s).

The ‘AIC12 EVM board can be used alone or installed on the development board. The EVM board contains one ‘AIC12 codec and the associated basic support circuits, similar to that shown in Figure 2. Additional circuits and connections on the EVM board allow for analog inputs and outputs that can be used to directly access various voice analog devices, such as microphone, speaker, earphone, and handset.

Configure the M/S and FSD pins for logic low when using the ‘AIC12 EVM board in codec stand-alone slave mode. See Table 2.

2.4 System Hardware Connection

To complete the ‘AIC12 EVM system, connect the ‘AIC12 EVM board to the AIC development board through the available connector. Install the development board on the DSK board using the 80-pin EPI connector. This provides a three-layer hardware system with the ‘C5402 DSK board as the mother board. The AIC/DSP development board is the daughter board, and the ‘AIC12 EVM board becomes the granddaughter board. Ensure all boards have been properly set up, as described above.

For developing the driver and applications shown in this report, the EVM system was connected to a PC and power supply. Figure 3 shows the AIC/DSP system and hardware connections.

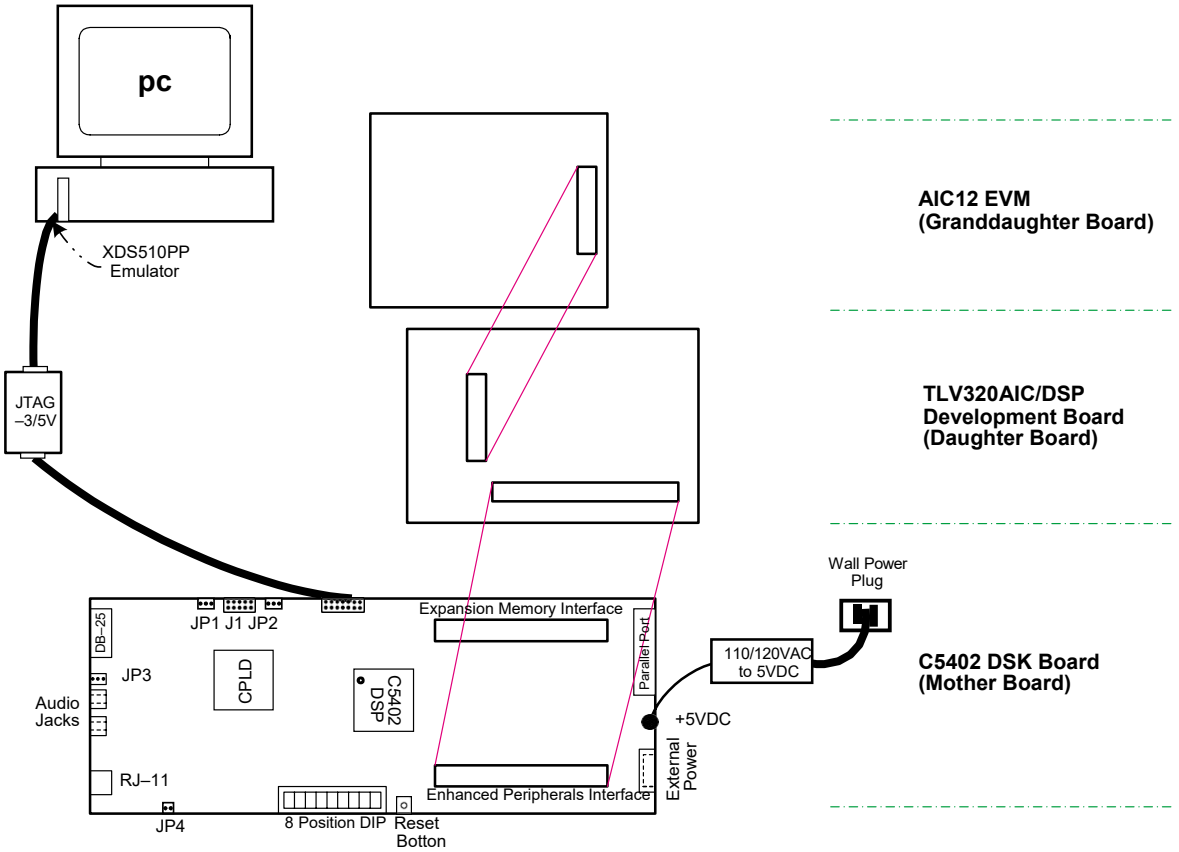


Figure 3. AIC/DSP EVM System Connection

3 Using TLV32012/13/14/15 Codecs

The data manual for the ‘AIC12 [1] identifies the features, performance, operation methods, digital port interface protocols and timing, definition for the control registers, and other important information. All of this is helpful and necessary for understanding and using the device. In practice, however, there is always something that needs to be emphasized and further clarified. This section provides helpful information to assist the user in finding and resolving some of these topics—especially for the AIC stand-alone slave applications.

3.1 Power Down and Reset

The ‘AIC12 can be powered down or reset by hardware or software, as illustrated by the data manual [1]. Tables 5 and 6 outline the functions, methods, and other items of importance. Note: CR stands for *Control Register* and additional information is given in the next section. Carefully follow the directions and procedures when using the power-down function, the reset function, or both.

Table 5. ’AIC12/13/14/15 Power Down

	HARDWARE POWER-DOWN	SOFTWARE POWER-DOWN
Function (and effect)	1) Entire ’AIC12 in power down 2) CR contents are preserved 3) Output at 3-state	1) There are three levels of power down 2) Digital interface is not powered down 3) CR contents are preserved 4) CRs accept control frame via DIN 5) Output at 3-state
Enable/ Disable Method	Logic low at pin #8 $\overline{\text{PWRDN}}$ to enable power-down; Logic high at pin #8 PWRDN to return to normal function	D7 and D6 at CR#3 D7 D6 = 00 No power down 01 A/D power down 10 D/A power down 11 All power down except digital interface
Notes	• Signal to $\overline{\text{PWRDN}}$ must sync with MCLK	• After setup D7 and D6 at CR#3, it will come into effect in the next FS

Table 6. ’AIC12/13/14/15 Reset

	HARDWARE POWER-DOWN	SOFTWARE POWER-DOWN
Function (and effect)	1) Reset all CRs (to default state) 2) Clear all sequential circuits 3) During reset, serial port in 3-state	1) Reset all CRs (to default state) 2) Clear all sequential circuits 3) During reset, serial port in 3-state
Enable Method	Low-going pulse at pin#24 $\overline{\text{RESET}}$	D5 at CR#3 D5 = 1 Reset the AIC (D5 will automatically go back to 0 after reset, and its default status is logic 0)
Notes	• $\overline{\text{RESET}}$ should sync with MCLK • RESET should be low for at least 6 MCLKs long • SCLK should be continuously running during the non-power-up reset • Reset initialization will last for at least 132 MCLKs • On power up or coming out of power down, a hardware reset is required; and the RESET pulse must be low for at least 10 ms.	• It is suggested to software reset the device before the initial CR configuration

The power down and reset pins, $\overline{\text{PWRDN}}$ and $\overline{\text{RESET}}$, on the ‘AIC12 device are active low. If not used, ensure these pins are either pulled high or tied high to enable normal operation.

Consider the $\overline{\text{PWRDN}}$ and $\overline{\text{RESET}}$ pins as the *sync* power-down and *sync* reset signals. As stated in Tables 5 and 6, both need to be synchronized with MCLK, the main reference clock of the codec. The reset circuit in Figure 2 shows an example of how to generate the proper reset signal.

Pay special attention to the requirement for resetting the ‘AIC12 through the hardware $\overline{\text{RESET}}$ pin after a power down/up, or when returning to operation from a hardware or software power down. Note that the power-up hardware reset pulse must be at least 10ms duration to ensure a proper reset. Otherwise, the device may not enter its default condition and therefore not operate properly.

One more precaution: Keep SCLK running during non-powerup reset.

3.2 AIC12 Control Register Programming Procedures

The ‘AIC12 codec has six control registers (CR). Refer to the data sheets [1] for the CR definitions. The control registers can be programmed or controlled using the data port or the host port, and the contents of the CRs can be read by the DSP (or MCU). The CR program procedure is usually called the *software configuration,* whereas the *hardware configuration* requires physical placement of jumpers, switches, etc.

After a power up and a hardware reset on the $\overline{\text{RESET}}$ pin, the ‘AIC12 enters the default condition. Software configuration is needed to put the AIC into an operating mode other than the default mode, and the AIC12 CR programming firmware routine should be executed.

It is strongly recommended that users wait for 2 frames before the DSP starts writing to the ‘AIC12 control registers through the data port, or the host port. In stand-alone slave mode, special attention should be given to meet this requirement.

The initial CR program sequence is very important for obvious reasons in some cases. One example is when setting the AIC to the *continuous* mode. If the data port is the programming port, CR#1 should be programmed *after* all other CRs. Otherwise, the other control registers cannot be programmed after CR#1 is programmed and the AIC mode is changed from programming mode (default) to the *continuous* mode.

The software configuration sequence shown in Table 7 is used for this application report, and is suitable for many applications using the ’AIC12 stand-alone slave mode. It is not the only sequence that can be used, however.

Table 7. Working ’AIC12 CR Software Configuration Sequence

CONFIGURATION STEP	FUNCTION
1	Reset the AIC by set D5=1 at the CR#3
2	Read back from CR#1 (to reset possible overflow flags).
3	Configure CR#2
4	Configure CR#4
5	Configure CR#5
6	Configure CR#6
7	Configure CR#3
8	Configure CR#1

If the default setting of a control register is appropriate, no programming is required for that particular CR.

3.3 Stand-Alone Slave and Turbo Modes

Under the stand-alone slave mode, the ‘AIC12 data port does not generate the data Rx/Tx clock (SCLK) and frame sync (FS) signals for the DSP/codec communication. Instead, these signals are generated from the 4-wire serial port of the host processor, or the McBSP of the DSP.

The ‘AIC12 is designed so that, in standard mode, the codec does not check and synchronize the FS after it has detected and synchronized with the first FS. This may not be a problem if an ‘AIC12 works in the stand-alone master mode, or in cascade mode, since the master ‘AIC12 generates the FS based on its own sample rate. However, in the stand-alone slave mode—since the interface FS and the AIC sample frame are from two independent sources—it can not be stated with certainty that the digital interface synchronizes with the ‘AIC12’s internal sample rate. The interface FS is from the McBSP of the DSP, and the ‘AIC12 sample frame is determined by the ‘AIC12 internally.

It is better to configure a stand-alone slave ‘AIC12 codec to the *turbo* mode. In turbo mode, the ‘AIC12 core checks and synchronizes the FS signal at each frame to generate the proper synchronization between the codec and the interface. Refer to the ‘AIC12 data manual and Figures 4 through 7 of this application report for the different timing schemes in the standard and turbo modes. Table 8 also lists the characteristics of FS and SCLK for the standard and the turbo modes.

Table 8. Stand-Alone Slave Standard and Turbo Modes

CHARACTERISTICS	STANDARD MODE	TURBO MODE
SCLK range	SCLK ≤ 32 × 26000 = 832 kHz	SCLK ≤ 25 MHz
SCLK and FS relation	SCLK = FS × 32 (Programming Mode) or SCLK = FS × 16 (Continuous Mode)	SCLK > FS × 32

3.4 Programming and Continuous Modes

The ‘AIC12 can be operated in either the *programming* or the *continuous* mode. The default mode, after a power-up, is the programming mode.

In the programming mode, there are at least two 16-bit time slots in every interval between 2 FS pulses. One of the time slots, called the *data frame*, receives/transmits the ADC/DAC data. The other slot is the *control frame* that receives/transmits the configuration code inside the AIC control registers, or the configuration code to be written to the CRs. Refer to Figures 4 and 5 timing diagrams for the standard programming mode and the turbo programming mode, respectively.

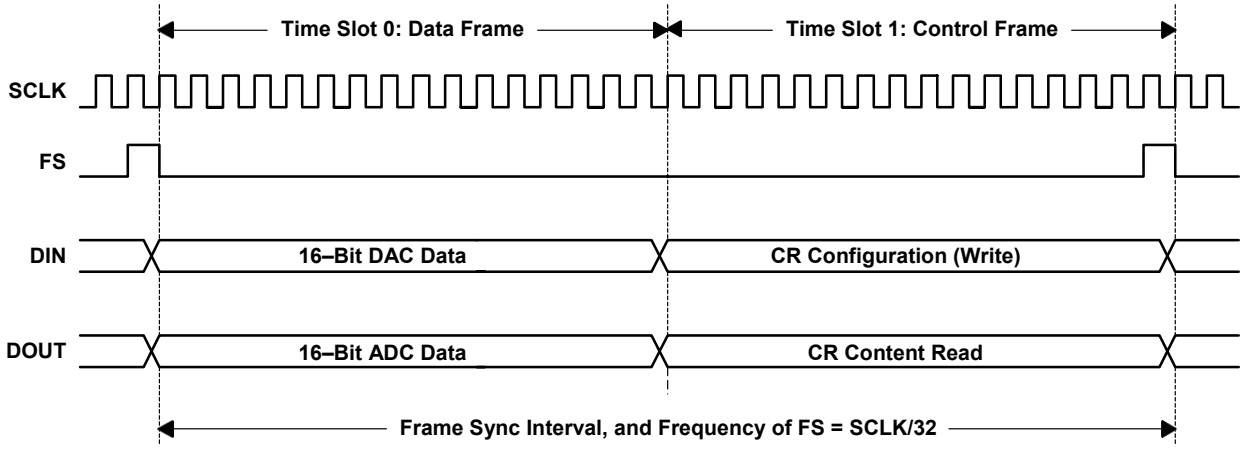


Figure 4. Standard Programming Mode Timing, Stand-Alone Slave

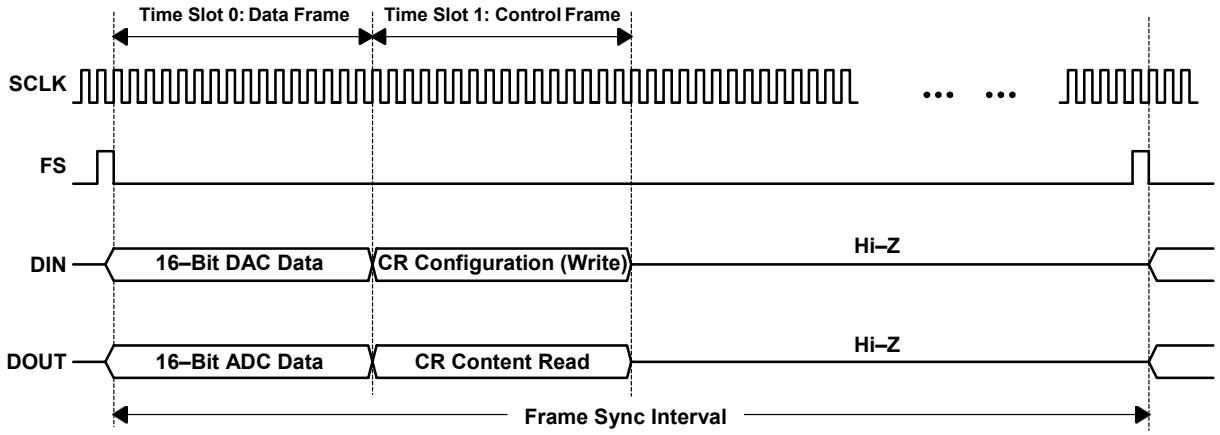


Figure 5. Turbo Programming Mode Timing, Stand-Alone Slave

It is very important to execute the control frames when the AIC is in the programming mode. The control frame cannot be skipped, even if no CR reading or configuration is actually needed. Make sure something is written, such as 0x0000, at each control frame.

No control frame exists in the continuous mode, as is shown in Figures 6 and 7 timing diagrams.

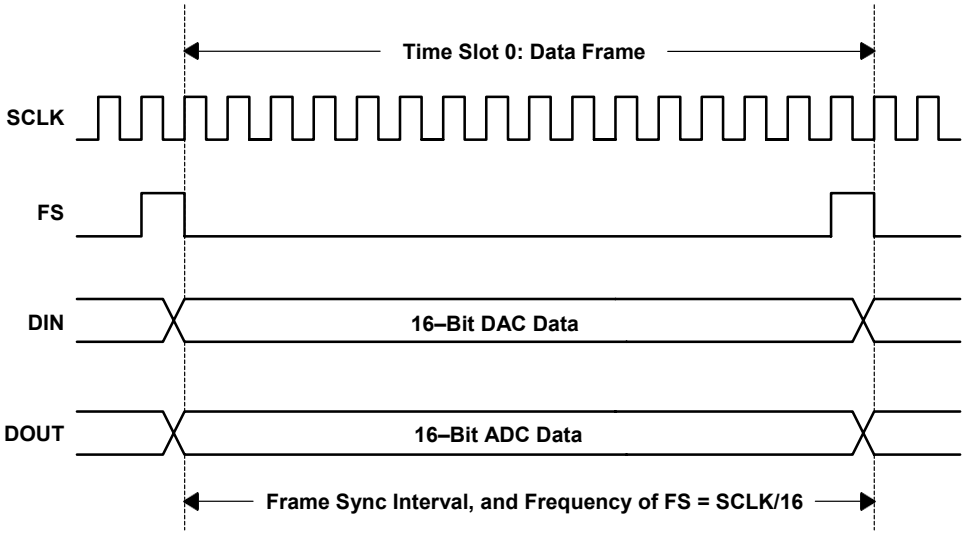


Figure 6. Standard Continuous Mode Timing, Stand-Alone Slave

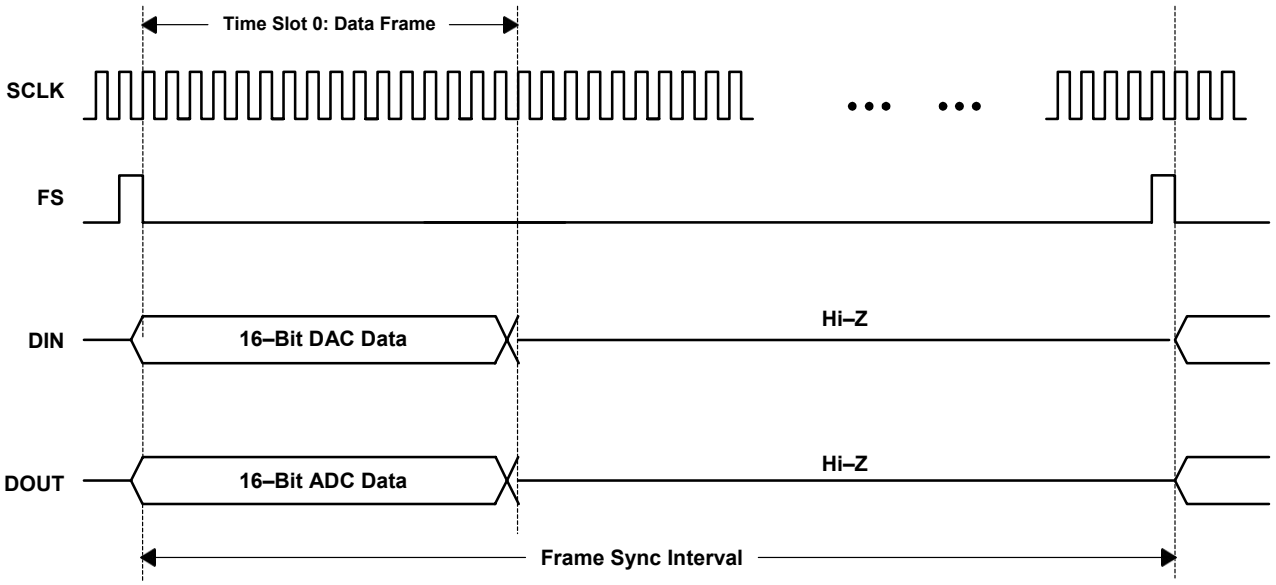


Figure 7. Turbo Continuous Mode Timing, Stand-Alone Slave

If the continuous mode is selected, either the two-wire host port is used for the AIC CR configuration (under I2C or S2C protocol), or the AIC must be operated under 15-bit data standard. In the latter situation, the D0 bit of the input data stream (through DIN) is used as a *control frame request*. If D0 is a 0, a normal continuous mode of operation is implied. If D0 is a 1, a control frame request is implied. This places the AIC back into the programming mode, allowing the host system to reprogram the AIC device. Refer to the 'AIC12 data manual for more information.

3.5 Conversion Rate Selection

An 'AIC12 device performs the ADC and DAC at the conversion frequency equal to the FS frequency. Internally, the frequency is generated from the master clock MCLK through three clock dividers, named M, N, and P, and:

FS Frequency = MCLK/16/M/N/P Hz

where M= 1, 2, … …, 128; N= 1, 2, … … 16; and P=1, 2, … 8.

When P=8, the conversion is referred to as *coarse sampling*; otherwise, it is *fine sampling.*

The master clock, MCLK, must be selected within the range of 10 MHz to 100 MHz when *fine sampling*. There are some other restrictions on the dividers. The divider selection criteria in stand-alone slave operation are:

1. FS = MCLK/16/M/N/P (where FS <= 26KHz);
2. 10 MHz <= MCLK/P <= 25MHz; in fine sampling;
3. M=1,2, …, 128;
4. N=1,2, …, 16;
5. P=1,2, …, 8;
6. The value of M should be a multiple of 2 if the DAC over-sampling rate is chosen for 256, and a multiple of 4 for 512. The DAC over-sampling rate is 128 by default and programmable by using D4 and D3 of CR#3.

7. The value of M should be a multiple of 4 if the ‘AIC12 internal IIR/FIR-filters were bypassed.

With a desired FS frequency and MCLK, there is usually more than one combination of the M, N, and P, that satisfies the criteria above. This gives users flexibility in obtaining the best combination for the desired application.

The dividers M, N, and P are selected and set up by configuring CR#4. At the power-up default the divider values in CR#4 are: M=16, N=6 and P=8. Hence, after a power up, the AIC FS frequency is MCLK/12288 Hz.

3.6 Differential or Single-Ended Analog Output

The ‘AIC12 can output its DAC result through three pairs of output pins. Analog output pair OUTP1/OUTM1 is the differential output. The other two output pairs, OUTP2/OUTMV, and OUTP3/OUTMV, can be used as differential or single-ended outputs, configurable through D7 in CR#6.

The differential outputs are better than the single-ended, in the sense that the common noise has less effect on the outputs. Moreover, if using the OUTP1/OUTM1 as the single-ended output (as shown in Figure 3-1 of the ‘AIC12 data manual [1]), the signal is only 1/2 compared to the differential output. Figure 8 shows example signals measured on the OUTP1 and OUTM1 pins—shown at the top and the middle plots of Figure 8. The differential signal (OUTP1–OUTM1) is also displayed at the bottom of the figure. Therefore, use OUTP1/OUTM1 in the differential mode whenever possible for significant signal-to-noise advantage.

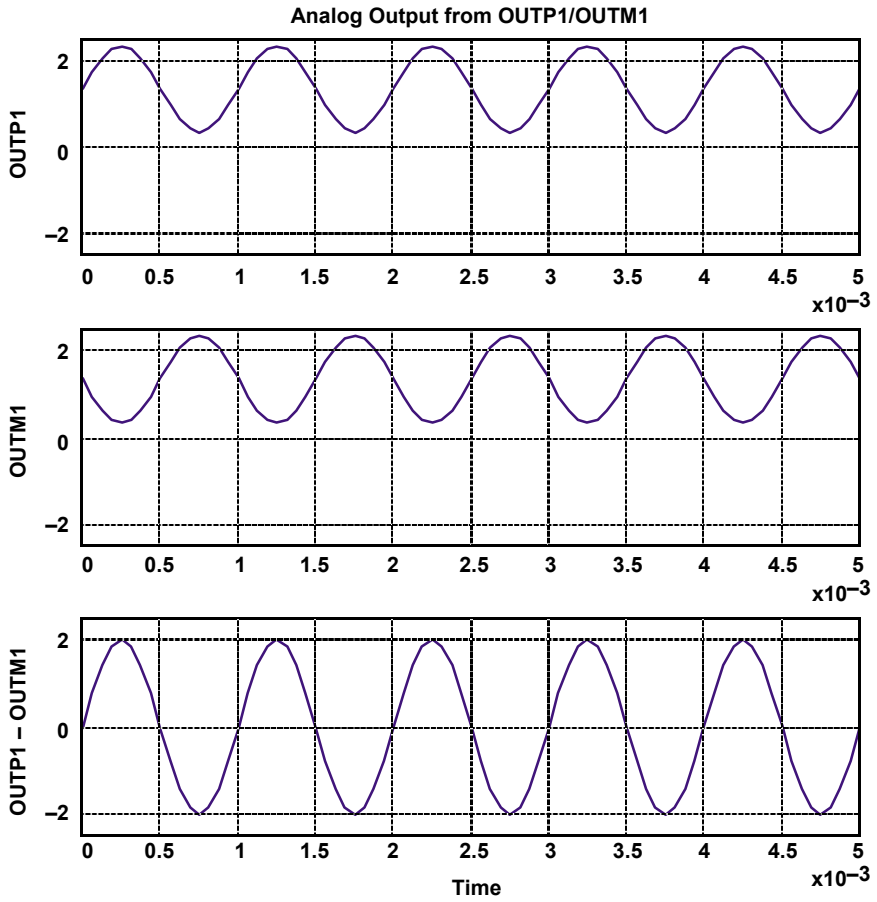


Figure 8. Single-Ended or Differential Signals at OUTP1/OUTM1

3.7 Power Supply Quality

Ensure the power pins (four pair) on the ‘AIC12 device are connected to the appropriate power sources. Refer to the Table of *Power Supply Pins* in Figure 1 for more details.

It is very important to have good quality power supplies. Refer to Section 5.10 in the data manual [1] to [4] for the power supply rejection restrictions. Additionally, when doing schematic design and PCB layout, follow appropriate design procedures for grounding and wiring. Separate the digital and analog grounds if at all possible.

The basic circuit for an ‘AIC12, shown in Figure 2, includes three capacitors on each of the digital power supply lines. *Do not ignore these capacitors.* Install the proper capacitors as close as possible to the device power pins.

4 Firmware System

Code for running the C54xx DSP is available online and colocated with this application report at the TI website. Information on DSP coding and the Code Composer Studio™ are also available online. See references [10] through [14]. The firmware has been tested and validated using the system configuration shown in Figure 3.

4.1 Firmware Design and Structure

The major objectives and functions of the firmware system are:

- A firmware structure that is reusable by different applications that incorporate a C54xx DSP and an ‘AIC12 codec running in stand-alone slave mode
- A software driver that interfaces between the ‘C5402 DSP and the ‘AIC12 codecs, including the software configuration and the data receive/transmit procedures
- Some simple applications or test routines that prove the basic system specification or functions

The developed code has been arranged into a simple structure, as the flow chart in Figure 9 illustrates.

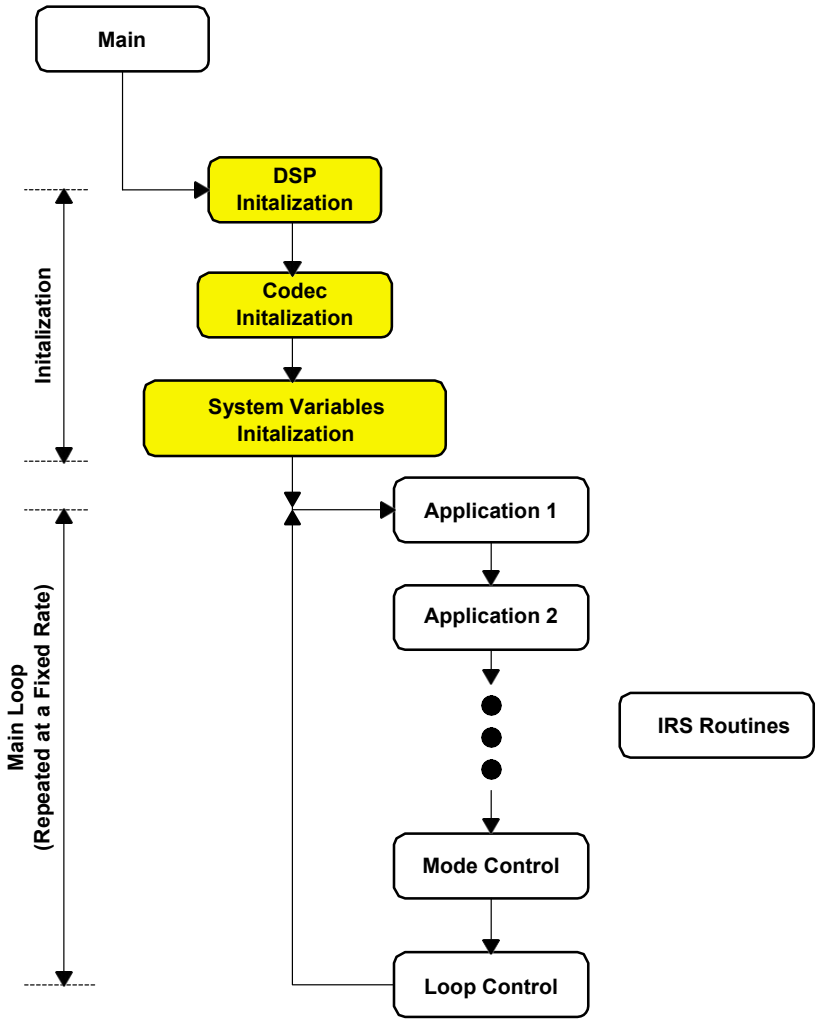


Figure 9. DSP/Codec Firmware Structure

On top of the structure, a firmware main routine calls all other routines for system initializations and applications.

The initialization routines are used to configure the DSP, the ‘AIC12 codec, and to set up the initial parameter values and variables. These routines bring the DSP/codec system into an initial working condition, ready to start the application routines. Refer to the ‘C5402 DSP initialization code, and the codec initialization code.

The customer-specified applications are run continuously, with a fixed repeating frequency within the main loop, and/or the interrupt service routines. Using this application report, along with the application examples and firmware, provides the user with these simple applications:

- Sine wave generator—a good way to test the DAC function and performance of the codec.
- DTMF tone generator—can be used for DAC and output driver circuit and device test.
- Signal loop back—routes the ADC input back to the DAC, testing both ADC and DAC paths of the codec, and the interface.
- Microphone system—same tests as the signal loop back, with input signal from a microphone.

The complete firmware code, with these application examples, is colocated with this application report in zip format, and can be down loaded by using the search engine at the TI web site:

www.ti.com

4.2 Configure ’C5402 DSP

The ‘C5402 configuration code has the following major fractional modulars:

- DSP CPU clock frequency setup
- Timer setup for the main loop controller
- McBSP setup for codec interface

The 20 MHz crystal oscillator on the DSK board provides a frequency reference for the ‘C5402 DSP. Three hardware clock-mode pins provide CLKMD1, CLKMD2 and CLKMD3, that divide or multiply the reference clock to obtain the default CPU clock for the DSP. After a power up, the CPU in the DSP runs at the default clock speed.

CLKMD, a memory-mapped register (MMR) in the DSP, can be used for displaying and changing the system clock and mode. The CLKMD address is at 0x58 in the MMR section of the DSP data memory. For this report, the CLKMD register is configured to 0x4007, indicating that the system/CPU clock frequency is PLLx5 = 100 MHz.

There are two general-purpose timers within a ‘C5402 DSP. For this application report. Timer0 controls the software main loop so that the application can be repeated at a specified fixed rate of 16KHz. To change the main loop to another running frequency, for example 8000Hz, simply load the period control register (PRD) of the timer with a value so that:

PRD = (CPU CLK frequency / main loop frequency) –1

Example:

= 100MHz/8KHz – 1

= 12499.

The main loop provides a time reference for other real-time applications, such as codec data receiving (Rx) and transmitting (Tx).

The ‘C5402 DSP has two McBSP ports. The AIC EVM system, as shown in the Figure 3, has McBSP0 port connected to the ‘AIC12’s host port and McBSP1 interfaces with the data port. Therefore, configure McBSP0 for implementing the S2I interface if the host port is utilized. Configure McBSP1 for voice data Rx/Tx (and for the ‘AIC12 programming, if the host-port is not used for reading/writing to the codec).

4.3 Configure ‘AIC12 Codec

This application report uses the stand-alone slave ‘AIC12 configuration routine for programming the codec though its data port. Refer to the the codec initialization code.

Before initialization, all codec control registers (CR) are at default values. The stand-alone slave codec is running at the programming and standard (non turbo) operating mode, and the data conversion rate is 50M/12288 = 4069Hz. Mode timing is shown in Figure 4.

Following the sequence given in Table 7 and applying the time frames shown in Figure 4, the DSP configures the control registers of the codec through the data-port to bring the device to its operating condition. The ‘AIC12 configuration routine puts the codec device into the programming turbo mode, with timing as shown in Figure 5. The conversion frequency or FS equals 7812.5 Hz (50M/16/25/4/4).

Additionally, the stand-alone slave codec can be placed into continuous turbo mode if D6 of control register CR#1 is set high. The digital inferface timing can be found in Figure 7. Note that turbo mode SCLK frequency does not need to be higher than continuous mode SCLK frequency.

4.4 Data Receive (ADC) and Transmit (DAC) Procedure

Several different methods can be used for data Rx/Tx between the McBSP and the codec: the main loop, the DMA of the DSP, or one or more of the DSP interrupts. For this application report, the interrupt method was used for data transfer.

Two interrupt vectors are employed in this application for moving the voice data. Refer to routine *C5402VEC.asm* in the available code. One vector is the McBSP1 receiving interrupt (BRINT1), set to be triggered whenever the McBSP1 receives 16–bit data, and the Rx ready flag is set. Another vector is the McBSP1 transmitting interrupt (BXINT1), set to be triggered by the Tx frame sync (FSX) signal.

There are two interrupt service routines (ISR), each responding to an interrupt vector. The ISRs are the *InstrSrvcT.c* routines in the code.

McBSP1 Rx interrupt occurs whenever a 16-bit data group from the codec is received by the McBSP1. Within the interval of two FS pulses, McBSP1 receives two 16-bit data groups, one is the ADC value of the ‘AIC12; the second is the CR content of the ‘AIC12. Refer to DOUT in Figure 4.

The ISR corresponding to BXINT1 is used to detect FS, or the first frame (also called *time slot 0*). At the first time slot, BRINT1 downloads the ADC data to memory, and uploads the digital data for the ‘AIC12 DAC. At the second time slot, BRIN1 downloads the ‘AIC12 CR content and sends a request for reading the next CR, or writing/configuring a control register.

The uploaded digital data (from the DSP) to the ‘AIC12 codec can be the signal generated or processed by the DSP. Running the application for this report, the single frequency sine wave or the double-tone-multiply-frequency (DTMF) tone is generated and sent to the ‘AIC12. By connecting an external analog signal generator or a microphone to the codec input, the ‘AIC12 converts the analog signal to digital and sends the ADC data to McBSP1. The BRINT1 ISR loops the digital signal back to the DAC path of the ‘AIC12. Refer to the code for the ISR routine.

5 References

1. TLV320AIC12, Low Power CMOS, 16-bit, 26-KSPS Codec with Smart Time Division Multiplexed (SMARTDM™) Serial Port, Data Manual (SLWS115)
2. TLV320AIC13, SMARTDM™ Low Power, Low Voltage, 1.1V to 3,6V I/O, 16-bit, 26-KSPS codec, Data Manual (SLWS139)
3. TLV320AIC14, Low Power CMOS, 16-bit, 26KSPS Codec with Smart Time Division Multiplexed (SM™) Serial Port, Data Manual (SLWS140)
4. TLV320AIC15, SMARTDM™ Low Power, Low Voltage, 1.1V to 3,6V I/O, 16-bit, 26-KSPS codec Data Manual (SLWS141)
5. TMS320VC5402 Fixed-Point Digital Signal Processor (SPRS079D)
6. TMS320C54x DSP CPU and Peripherals, Reference Set Volume 1 (SPRU131F)
7. TMS320C54x DSP Enhanced Peripherals, Reference Set Volume 5 (SPRU302)
8. TMS320 Cross-Platform Daughtercard Specification (SPRA711)
9. Codec Evaluation System (SLAA141)
10. Code Composer User’s Guide (SPRU296)
11. TMS320C54x DSP Mnemonic Instruction Set, Reference Set Volume 2 (SPRU172B)
12. TMS320C54x Assembly Language Tools, User’s Guide (SPRU102D)
13. TMS320C54x Optimizing C Compiler, User’s Guide (SPRU103D)
14. TMS320C54x DSP Applications Guide, Reference Set Volume 4 (SPRU173)

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