Synchronizing Multiple Parallel Devices Using the Internal PLL

Stephen Maggiotto   *Audio Converters*

ABSTRACT

The Texas Instruments family of audio converters can be used in various multichip configurations in order to realize multichannel processing using the time-division multiplexing (TDM) feature. Using a multichip solution sometimes requires synchronization of channels such that all of the devices operating in parallel operate in lock-step with each other to prevent phase shift errors between channels. The mechanisms and programming of the TLV320ADC3101 and TLV320ADC3100 audio ADCs are discussed in this application report.

1   Introduction

Each application for multichannel support may have different requirements having to do with selection of master clocks and synchronization. This application report specifically addresses the usage of the on-chip PLL of the ADC310x audio ADC to generate the audio serial interface clocks from a master clock (MCLK) reference, as well as the programming and sequencing of instructions such that all of the parallel devices are instruction locked. The method and programming instructions used in this application report were tested using three TLV320ADC3101 evaluation modules (EVM) and its companion USB motherboard. Verification of this method and programming was performed using an Audio Precision test instrument by measuring the phase difference between each channel. The configuration in this application report uses three devices in parallel. This is an arbitrary choice and can be expanded to as many devices as needed by the application.

2 Clock Configuration Block Diagram

Figure 1 shows that Device #1 is the master of the audio serial interface bus, while devices 2 and 3 are slaves. Device #1 takes in a master clock as input which is used as reference by the internal PLL, which in turn produces the bit clock (BCLK) and word clock (WCLK) that are used by the other devices as the audio serial interface clocks. The configuration of the audio serial interface clocks in the master device has to be done so the bit clock provided to the slave devices is fast enough to accomplish the processing requirements for each part.
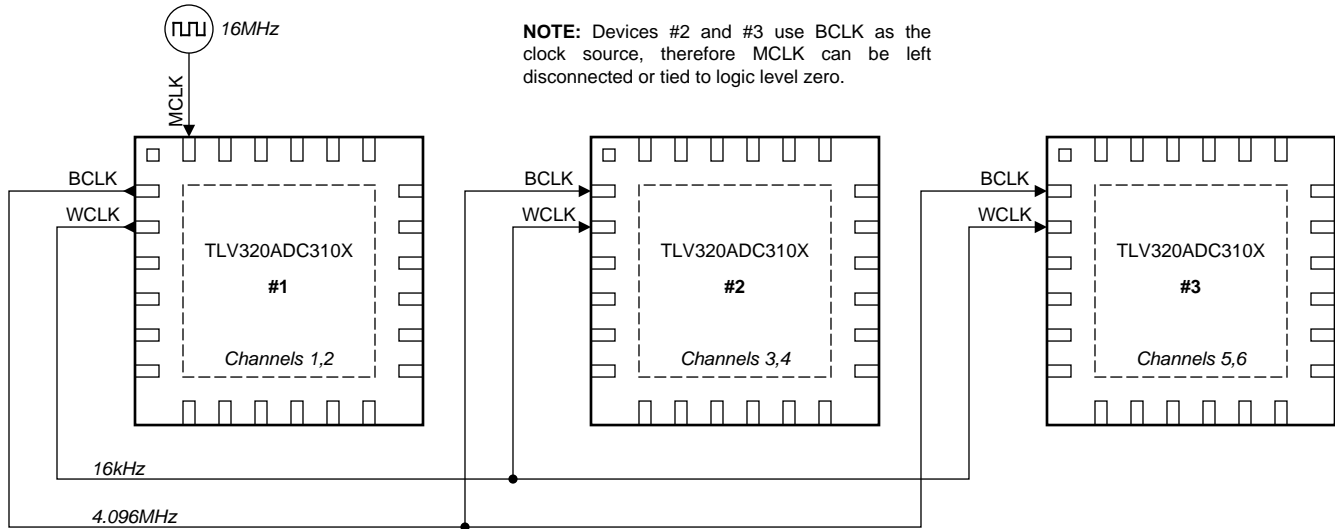

Figure 1. Multiple TLV320ADC310x Device Configuration

3   Conceptual Programming Sequence

The sequence of programming parallel devices depends on the PLL running and getting stable as the first step, followed by the clock configuration of the slave devices. After this, the programming of all the devices according to their required processing should be done. Then, clocks are released simultaneously by the master to all the slave devices, followed by the setting up of the TDM channels and the enabling of the respective DOUT pins. This method ensures that all of the devices' instruction counters are in lock-step with each other. Section 5 of this application report contains The Detailed Programming Sequence description. Figure 2 is the instruction sequence flow diagram.
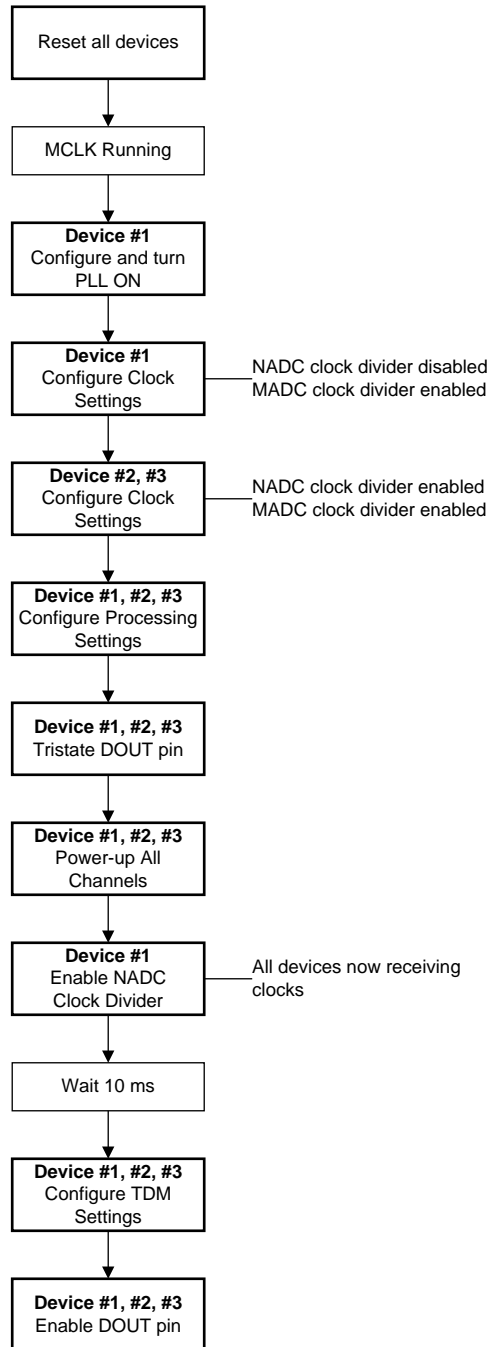

Figure 2. Instruction Sequence Flow Diagram

4   Other Important Considerations

The serial interface frame rate, defined by WCLK, has to be equal to the ADC sample rate or the user may experience repeat or missing samples. Setting up the same ADC miniDSP processing and associated clocks across all the devices is critical to capture waveforms with no phase difference in a multichip configuration. Regardless of the source for the master clock reference, MCLK must have a frequency that, after dividing by certain variables, as shown in the following text, is equal to the sampling rate of the audio serial interface.

(BCLK frequency)/(WCLK frequency) must be an integer.

ADC output sample rate = (ADC_CLKIN frequency)/(NADC×MADC×AOSR) = WCLK frequency

Where:
>   ADC_CLKIN refers to the internal ADC system clock, as referenced in Figure 28 of the TLV320ADC3101 datasheet.
>   NADC = Divider for the ADC MAC engine clock
>   MADC = Divider for the ADC modulator clock
>   AOSR = ADC over-sampling ratio

When the PLL is powered on, a start-up delay of approximately 10 milliseconds occurs after the power-up command of the PLL and before the clocks are available to the TLV320ADC3101. This delay ensures stable operation of the PLL and clock-divider logic. If the I2C programming sequence is faster than the mentioned delay time, then it is possible that the sequencing of events will not occur as expected after the PLL clock is available. This may lead to unexpected results, therefore, the user must ensure that the programming sequence is longer than this time delay, or in the event that it is not, add a delay such that the enabling of the NADC divider for Device #1 occurs after the PLL clock is available.

5   Detailed Programming Sequence

The following set of instructions follow the I2C format which has the first byte being the device address, followed by the register address byte and then the data byte. The I2C device address is a 7-bit value followed by a Read/Write bit which forms the address byte. Because all of the following listed commands are writes, the eighth bit is always logic level 0. As an example, the 7-bit device address for Device #1 is 0x18. When the eighth bit is added to it, in this case it is a logic level 0, the address byte becomes 0x30. If the user chooses to do a Read for Device #1, the address byte becomes 0x31. Because the ADC3101 is limited in the programmability of its I2C address (I2C_ADR1, I2C_ADR0), an I2C Expander device is needed if more than four devices were required to operate in parallel. here

Write Address Byte for Device #1 = 0x30 (7-bit Device Address is 0x18)
Write Address Byte for Device #2 = 0x32 (7-bit Device Address is 0x19)
Write Address Byte for Device #3 = 0x34 (7-bit Device Address is 0x1A)

```
###############################################
# Software Reset
###############################################
30 01 01
32 01 01
34 01 01


###############################################
# Device 1, Master Mode, PLL input frequency is 10 MHz
# WCLK Input Frequency is 16 kHz
# Configure PLL to produce 81.92 MHz
###############################################
30 04 03
# J=8
30 06 08
# D=1920 (multi-byte write)
30 07 07 80
# Enable PLL; P, R = 1
30 05 91
# Divide-By for NADC = 20; DISABLED
30 12 14
# Divide-By for MADC = 2; Enabled
30 13 82
# Serial Interface Configured; DSP mode, BCLK & WCLK are Outputs
30 1b 4d
# Source of BCLK is ADC_CLK; Divide-By = 1
30 1e 81


###############################################
# Clock Settings For Devices 2 & 3
# The CODECs receive: MCLK through BCLK at = 4.096 MHz,
# BCLK = 4.096 MHz, WCLK = 16 kHz
###############################################
# Select Page 0
32 00 00
34 00 00
# Divide-By for NADC = 1; Divide-By for MADC = 2; Enabled (multi-byte write)
32 12 81 82
34 12 81 82
# AOSR = 128 (default)
# Source of MCLK is from BCLK
32 04 01
34 04 01


###############################################
# User Defined ADC and Processing Settings
###############################################
Not covered here as they are application specific.


###############################################
# Enable NADC Divider of Device #1
###############################################
# Divide-By for NADC = 20; ENABLED
30 12 94
```

Before proceeding with the next set of instructions, the user must wait approximately 100 ms to allow
all of the parallel devices time to propagate the master clock throughout their logic cones

(i.e., clock dividers, mini-DSP, clock-gating logic,..etc).
  Insert Delay

```
#################################################
# TDM Channel Settings
# The settings below are arbitrary. User can select
# whichever slot assignment the system requires.
#################################################
# Select Page 0
30 00 00
32 00 00
34 00 00
#################################################
# Configure TDM Settings and Early 3-State
#################################################
30 26 03
32 26 03
34 26 03
# Channel 1 Offset
30 1c 00
# Channel 2 Offset
30 25 00
# Channel 3 Offset (Device #2 is offset from Device #1 by 32 clocks)
32 1c 20
# Channel 4 Offset
32 25 00
# Channel 5 Offset (Device #3 is offset from Device #1 by 64 clocks)
34 1c 40
# Channel 6 Offset
34 25 00
#################################################
# Enable DOUT Pins
#################################################
30 35 12
32 35 12
```

6   Cautionary Note

The preceding programming sequence is intended to work with the built-in filter modes because they output data from the miniDSP to the serial interface near the end of the instruction sequence. In other words, the device programmed to be in the last time slot starts transmitting serial data before the DSP output happens such that each device transmits the same sample in the same frame (i.e., in terms of time index).

7   References

1. Texas Instruments, *TLV320ADC3100 Low-Power Stereo ADC for Voice-Activated Systems and Portable Audio* data sheet
2. Texas Instruments, *TLV320ADC3101 Low-Power Stereo ADC With Embedded miniDSP for Wireless Handsets and Portable Audio* data sheet
3. Texas Instruments, *TLV320ADC3101-K* user's guide

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.