Freescale Semiconductor
Application Note

Document Number: AN4944
Rev. 0, 04/2013

Emulating I2S bus on Kinetis-M

Audio codec IC application example

by *Michael Galda*

Contents

1 Introduction

This example shows an atypical usage of the Kinetis-M (Metering) MCU family, which is based on a precise Sigma-Delta ADC and ARM CM0+ core in the role of the Audio Codec IC.

The purpose of this application note is to demonstrate that the Sigma/Delta-ADC analog front end (AFE) is effective at processing audio signals. The KM device uses two ADC channels to sample the stereo audio signal at 48.000 kHz with 24-bit precision. Then audio samples are transferred by DMA to the SPI module output, which in connection with the timer and peripheral crossbar module (PXBAR) emulates the I2S audio bus functionality, with no CPU loading.

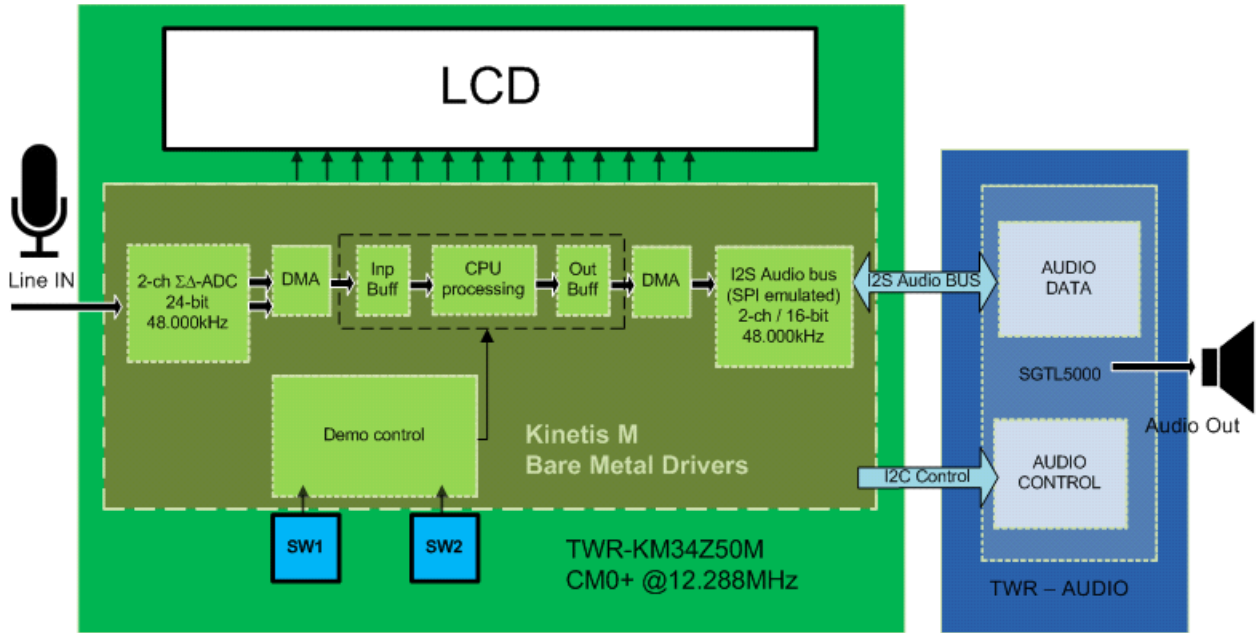

Figure 1. Application implementation on the Tower system

2 Required hardware

This document describes the application based on the Freescale Tower system, but the basic concept can be easily reproduced on the customized hardware as well.

The application can be set up using the following Tower system boards:

- TWR-KM34Z50M
- TWR-AUDIO-SGTL5000
- TWR-PROTO (with additional wiring connections)
- TWR-ELEV (primary and secondary)

The KM34 device acts as the I2S bus master, producing all required signals:

- Master Clock (MCLK = 12.288 MHz)
- Word Select (WS/LRCLK = 48.000 kHz)
- Continuous Serial Clock (SCK/SCLK = 1.536 MHz)
- Serial Data (SD/DOUT)

The KM34 MCU can set up and control the SGTL5000 codec settings via I^2C bus.

3 Emulating I2S bus signals - theory

The I2S is an audio bus using a three-wire connection for synchronous serial data communication.

Data are transmitted on the SD (DOUT) line (MSB first) in little endian format. Data length is 16-bit. Transmitter data are synchronized on rising edge of SCK and receiver data on the falling edge of SCK. A two-channel audio signal is represented by two data words, the right and the left channel sample, transmitted and multiplexed on the same wire. The word select (WS) control signal determines if the word is for the right or left channel. This signal also denotes the data length (the beginning and the end of the word). WS can be synchronized to either the rising or the falling SCK edge and precedes the MSB by one SCK period in order to have enough time to store the data by the receiver.

The main restriction of the first KM device is that there is no support of I2S HW module on silicon. There is a simple hardware workaround to overcome this limitation.

In this workaround we use SPI to generate the bit clock (SCK) and shift out the stream of audio data. A missing word select (WS) signal is generated by the timer output connected to the output pin via peripheral crossbar switch (PXBAR), while the timer input counts the SCK edges. This means that the timer input must be externally connected to the SPI clock output.

When the timer counter counts to 16 SCK edges, the counter is reloaded and the counter output is toggled, generating the WS output signal. Master Clock (MCLK) output is derived from the internal BUS clock, which can be routed to external MCU pin by software settings. In this configuration all emulated I2S signals are synchronous to the internal bus clock with the minimal jitter. Effective usage of the DMA channels for reading the samples from SD-ADC and feeding the SPI output data register ensures smooth audio data stream output without any interruption and with minimal CPU load. The remaining free CPU cycles can be used to do simple audio signal processing like IIR/FIR filtering (Lo-Pass, Hi-Pass), FFT and effects.

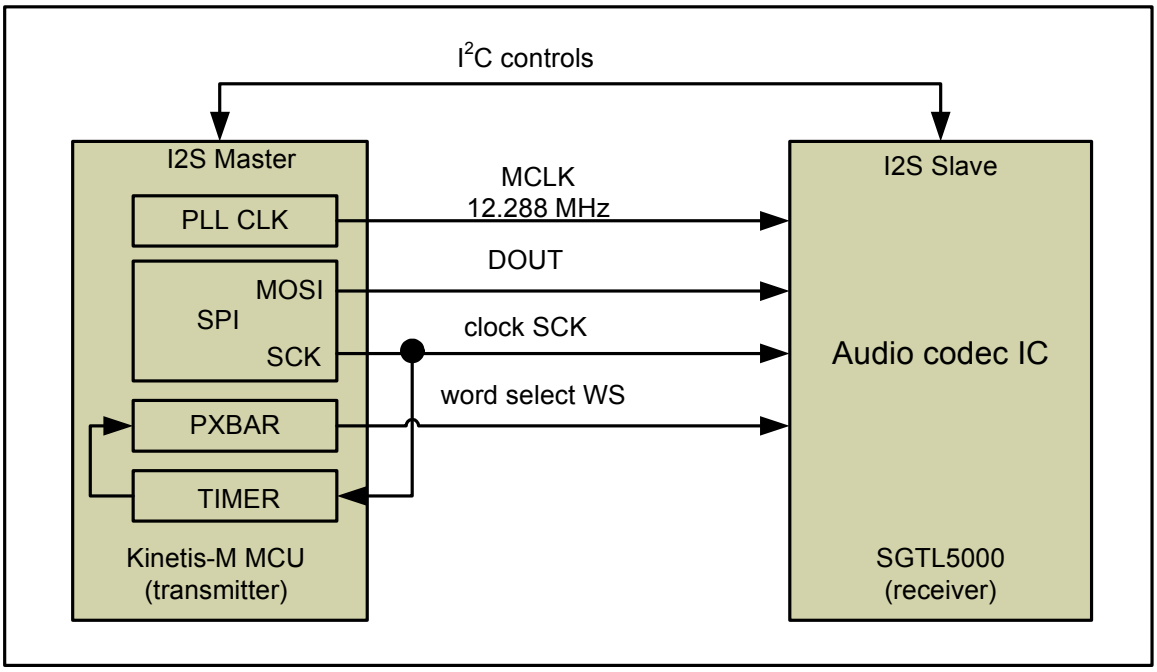

Figure 2. Master - Slave with emulated I2S

As mentioned above, due to the WS (word select) signal emulation, the SCK out must be connected externally to timer input providing a feedback path counting the clock edges. The WS signal is toggled after every 16-SCK edges to split the data valid for the left and right audio channel. The WS signal is generated by timer output. Because the QTIMER module used, Kinetis-M family supports only a single external pin connection for one channel (acting either as timer input or output). PXBAR (peripheral crossbar switch) can route the timer channel output internally to one of the PXBAR output (XBAR_OUTn) pins. Alternatively two timer channels can be concatenated in a cascade-count mode so that the counter's output is connected to the input of the 2nd timer channel.

The audio samples are reproduced by a SGTL5000 audio codec IC on the TWR-AUDIO-SGTL card. The SGTL5000 works in the I2S Slave mode. Any other I2S audio codec IC can be used for reconstruction of the audio signal on the customized HW.

The following table summarizes the required wired connections, which must be additionally created on the TWR-PROTO board, or header J10 of TWR-KM34Z50M. TWR-PROTO signals noted in bold must be interconnected.

Table 1. TWR-PROTO signal connections for I2S emulation

I2S signal	TWR-KM34Z50M		TWR-PROTO	TWR-AUDIO-SGTL5000	TWR-PROTO
MCLK out	CLKOUT	PTF7	**A33**	I2S0_MCLK	**A21**
TX_BCLK	SPI_SCK	PTF4	**B48**	I2S0_SCK	**A22**
LRCLK(WS)	PXBAR_OUT0	PTA7	**A51**	I2S0_WS	**A23**
TX_DATA	SPI_MOSI	PTF6	**B45**	I2S0_DOUT	**A25**
Clock counter	QT1	PTG0	**A40**	—	**B48**
Clock feedback	SPI_SCK	PTF4	**B48**	—	**A40**

4 Connecting the audio signal to the SD-ADC input

The main advantage of the AFE on the Kinetis-M family is the differential ADC input which can eliminate the input noise accumulated on the analog wiring. However, for the basic testing and evaluation of this application note, a single ended connection has been used. The SDASP0,1 inputs are connected to L and R signals and SDADM0,1 are connected directly to VSSA (analog GND). In this single-ended connection, the input range for the externally generated audio signal is +/- 500mV (1V pk-to-pk), because the SD-ADC can measure a negative voltage against the SDADM pins. Simple RC lo-pass filter with the cut-off frequency of about 24 kHz has been selected. The input impedance of the SD-ADC is between 30-50k Ohms depending on the settings. So that any external audio signal source with the appropriate LINE-OUT levels can be safely connected directly to SD-ADC inputs. The audio signal wiring can be managed on the TWR-PROTO board or better directly interfaced to J17 on TWR-KM34Z50M.

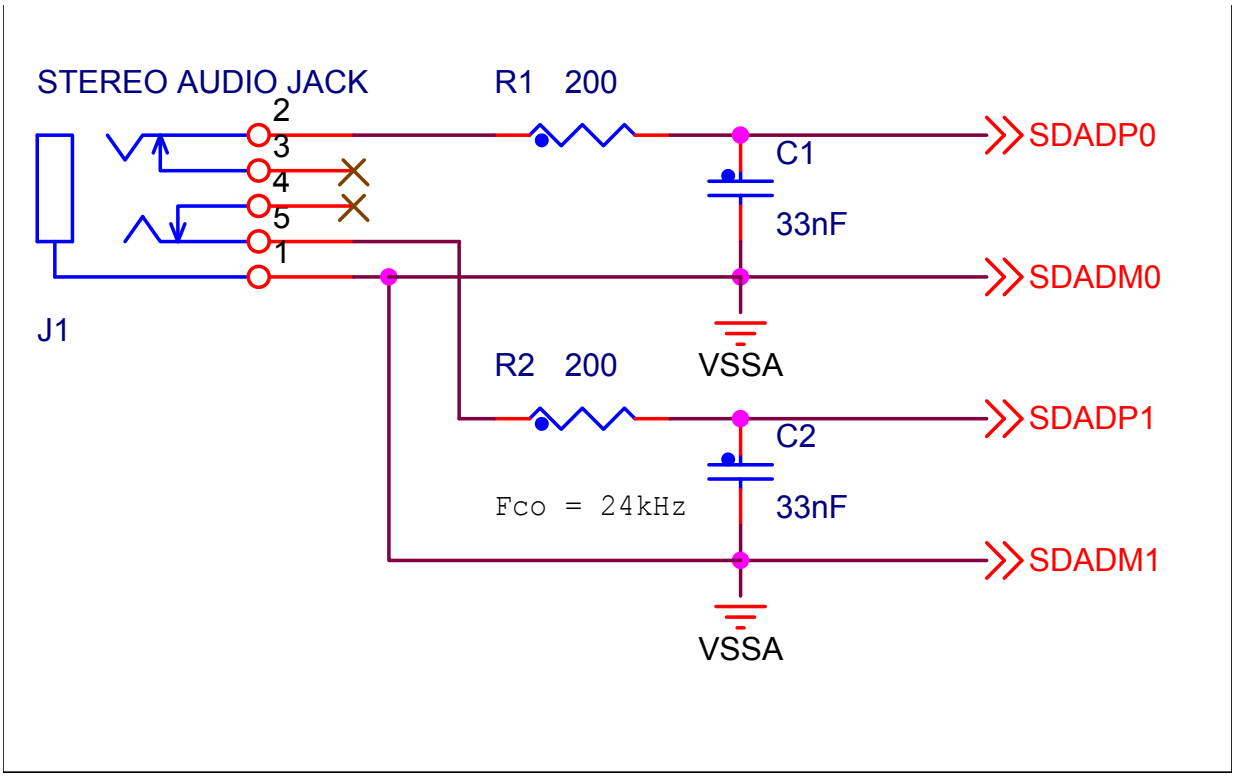

Figure 3. Recommended external audio signal connections to SD-ADC

5 SW description

The software is based on Kinetis-M Peripheral BareMetal Drivers EAR2.2, available for free download on www.freescale.com. These low level drivers allow rapid software development. The software example is titled “AN4944_SW.zip”.

NOTE

The software has been written to run on the TWR-KM34Z50M with no additional hardware modifications. There is a CPU performance limitation caused by metering use cases oriented to Tower card design. On the Tower card, the core clock is derived from the 32.768 kHz oscillator multiplied by PLL x 375, providing the 12.288 MHz for the AFE module and the system. This is the typical metering application requirement. Deriving the clock from an on-board 8 MHz crystal by FLL is not recommended due higher clock jitter. For the best performance, the MCU clock can be supplied by an 24.576 MHz or 49.152 MHz external crystal or oscillator (typical audio applications crystals). Operating the CPU at higher frequencies adds cycles to the signal processing.

5.1 Initial settings

In the init, all required peripheral module clock are enabled in SIM and the internal PLL is enabled running the BUS clock at 12.288 MHz. The bus clock is routed to pin PTF7, which serves as I2S MCLK (master clock) output. The clock ratio between the system: bus: flash is then set to 1:1:1.

Please note that for the system clock higher than 25 MHz, the ratio must be set to 1:2:2 due to the flash clock frequency limitation to 25 MHz.

The SYSTICK (core timer) is initialized just for the CPU loading measurements to count the CPU cycles spent for servicing the IRQs and required for signal filtering.

Then the GPIO pins are switched to the appropriate peripheral functionality and output level.

5.2 DMA configuration

Two DMA (DMA0, 3) channels are used for reading the data from two AFE channels giving the samples from the left and right audio channels. The samples are copied into the buffer in the internal SRAM. The third DMA channel (DMA2) is used for feeding the SPI data output by samples. After a certain block of samples is transferred the DMA channels are serviced and reloaded in the appropriate ISR routines.

5.3 AFE configuration

AFE clock is set to PLL clock divided by two to get the 6.144 MHz with the oversampling rate (OSR=128) resulting to the sampling rate F_s = 48.000 kHz.

5.4 I2S bus emulation in software

I2S bus functionality is emulated by the following mechanism, which ensures a smooth, continuous stream of audio data on the SPI output:

SPI is configured in 16 bit master mode, DMA channel (DMA2) fills the SPI output data register periodically with left and right audio samples from the buffer. SPI baud rate is set according to the desired audio, i.e. to 1.536 MHz = (48.000 kHz * 2ch * 16 bits).

Timer/Counter channel1 (TMR1) is set in up counting mode with auto-clearing after compare, counting the rising edges of the SPI CLK signal. The SPI CLK must be connected externally to timer input1 (PTG0).

XBAR0 (Peripheral Crossbar Switch) is configured to route the TMR1 output to XBAR_OUT0 (PTA7), the XBAR0 output is toggled when the counter counts 16 CLK edges, which serves for proper I2S WS signal generation. At the beginning the timer counter is preloaded with value 1, which causes the WS signal toggles before the last bit of the current word is shifted out on the bus. This means that the WS signal change precedes the new word by one SPI clock cycle. This is required by I2S bus specifications. The WS signal change denotes the last bit of the current word is being shifted out allowing the receiver device to latch the current word.

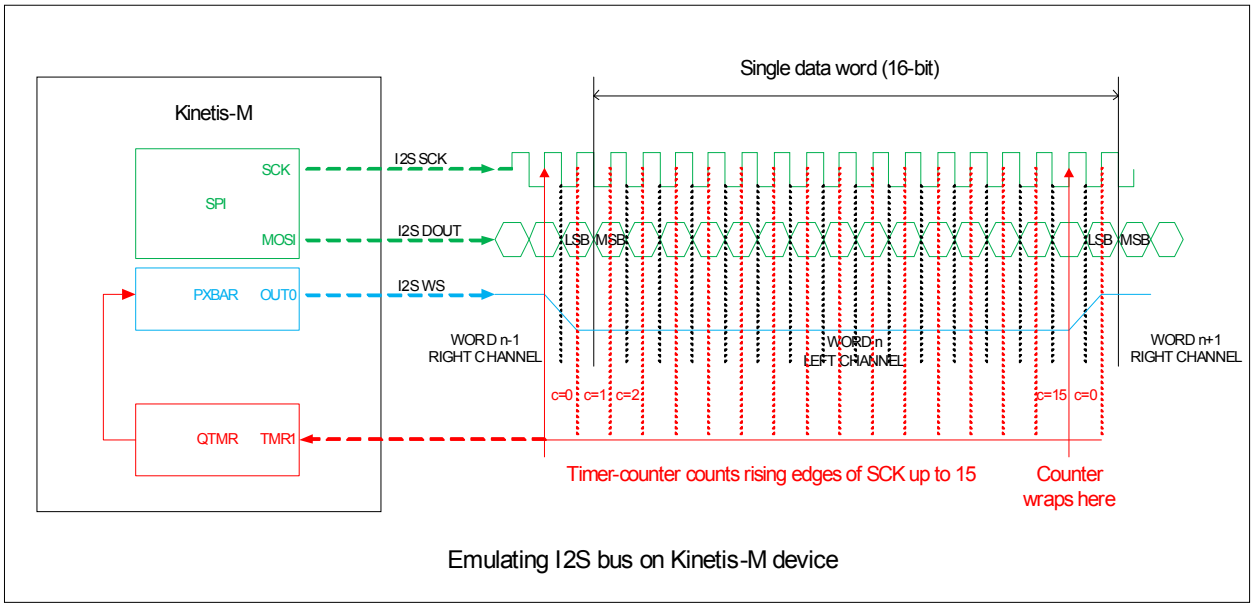

Figure 4. Emulating I2S bus signals on Kinetis-M device

5.5 Additional SW tasks

TWR-KM34Z50M supports a LCD display and two push buttons. These can be used for demo application control. Onboard LEDs can be used for input signal level indication. The application control is interrupt driven. Note that these application interrupts must have lower priorities than the interrupts responsible for audio stream generation to ensure non-intermittent audio data output.

5.6 Digital signal processing

Due to the offloading of the CPU by the DMA machine to transfer the data stream, the free CPU cycles can be used for basic signal digital filtering in the real time (FIR/IIR filters). Experiments indicate that the ARM CM0+ CPU running at 12.288 MHz is capable of calculating up to fourth order FIR filters for both audio channels (48.000 kHz stereo), which is sufficient for simple lo-pass/hi-pass filter for instance. Running the CPU core at higher frequencies (up to 50 MHz), we will get higher processing performance to calculate more complex filters.

5.7 Conclusion

This application shows the Kinetis-M family MCU used as simple audio codec IC, sampling the stereo audio signal by 2-channels SD-ADC. The I2S audio bus functionality is successfully emulated by SPI and timer/counter modules. Smart usage of the DMA channels can offload the CPU and save the CPU cycles for another software application tasks.

6 References

[1] I2S Bus specification – Philips (NXP) Semiconductors, 1986:

https://www.sparkfun.com/datasheets/BreakoutBoards/I2SBUS.pdf

[2] AN4309 Application note: Playing audio file interfacing STM32L1xx microcontroller with an external I2S audio codec – STMicroelectronics, 2013:

http://www.st.com/st-web-ui/static/active/en/resource/technical/document/application_note/DM00087544.pdf

[3] Freescale Tower system:

http://www.freescale.com/tower

[4] TWR-KM34Z50M: Kinetis M Series Tower System Module:

http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=TWR-KM34Z50M

Appendix A

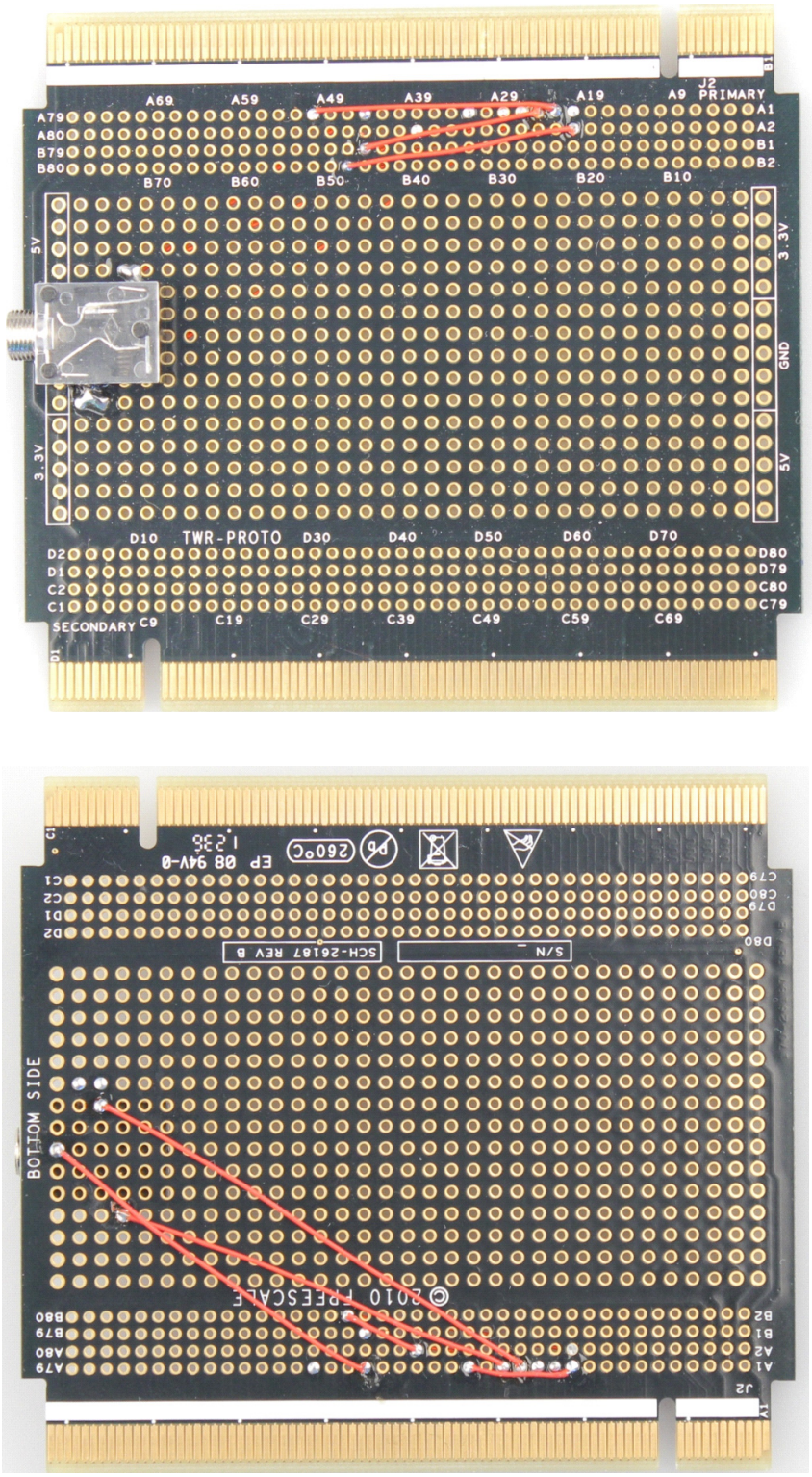

Figure 5. Additional signal connections made on TWR-PROTO card

Appendix B

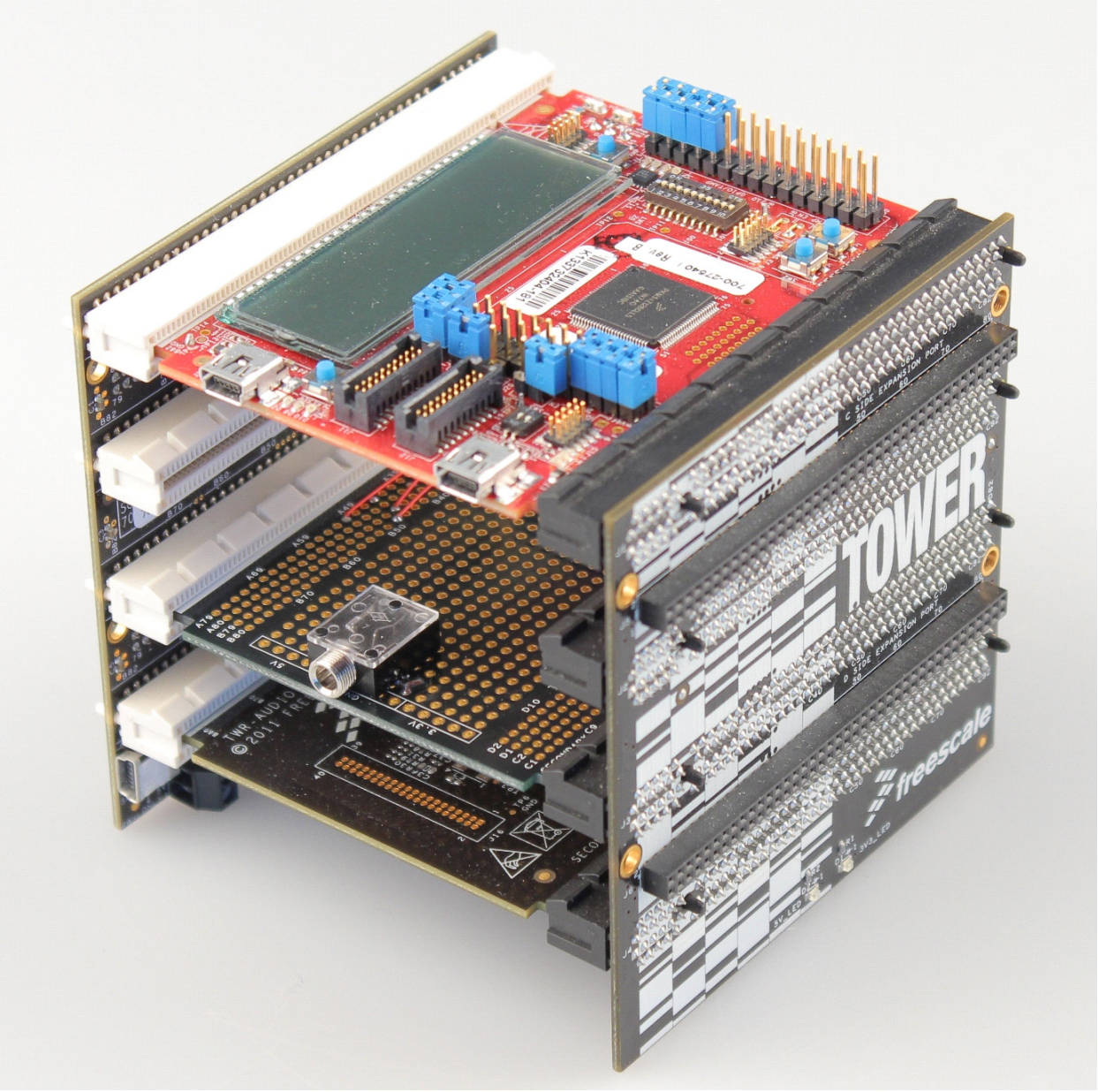

Figure 6. Demo application based on Freescale Tower system

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number: AN4944
Rev. 0
04/2013