UTILITY CLASS-D TOOLSTICK USER’S GUIDE

1. Introduction

Class-D is a switching power amplifier architecture that uses high-frequency Pulse-Width Modulation (PWM) to generate the output waveform. The transistors are fully on or fully off, which means this architecture can reach very high efficiency levels, resulting in significant power savings. However, the translation from the input signal to PWM and the PWM itself can cause more distortion on the output than other amplifier architectures. Figure 1 shows a very basic Class-D signal diagram.

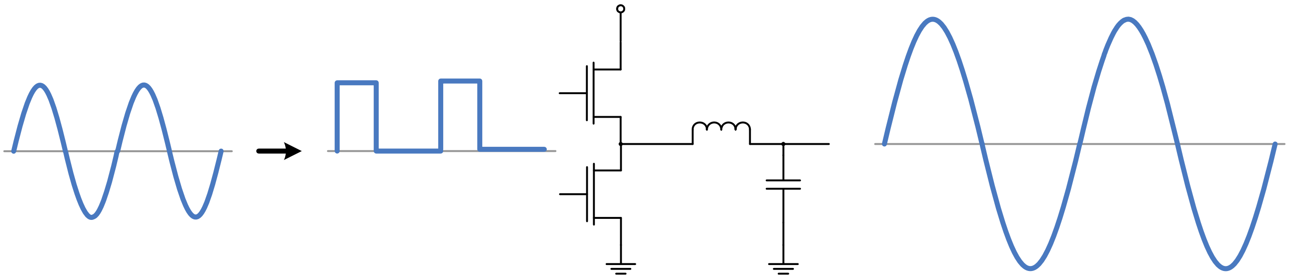


Figure 1. Class-D Power Amplifier Signal Diagram

Because the Class-D amplifier transistors are fully on or fully off, the transistors lose less energy in the form of heat. This architecture, therefore, lends itself to small, cheap MOSFETs. The Class-D amplifier can be implemented in either analog or digital form.

An analog Class-D amplifier is non-trivial and typically consists of a comparator, triangle waveform generator, and several blocks to condition the input signal before transferring to the output MOSFETs.

A digital Class-D amplifier requires the following:

- PWM output (switching) frequency 10x+ faster than the highest input frequency to adequately reconstruct the input signal
- High-resolution control of the PWM pulse width to reduce output quantization distortion
- Method for sampling or receiving the input waveform
- Fast core for digital processing and manipulation of data
- Pins capable of driving the amplified signal

Since the Silicon Labs Precision32™ SiM3U1xx/SiM3C1xx 32-bit MCUs have peripherals and features capable of meeting all of these requirements, these MCUs are uniquely suited to a utility Class-D power amplifier application without using many external components by directly driving the speaker using the high drive I/O.

In addition to providing a development platform for Class-D applications, the Class-D ToolStick board serves as a lower-cost, general-purpose development platform for the SiM3U1xx USB MCUs.

2. Relevant Documents

This document provides a hardware and software overview for the Class-D ToolStick. Additional documentation on the Precision32 tools and MCUs can be found on the Silicon Labs website: www.silabs.com/32bit-appnotes, www.silabs.com/32bit-software, and www.silabs.com/32bit-mcu.

3. Using the Demo

The speaker in the kit connects to the JP1 1x2 header next to the green terminal block (J4). Two alternate connection methods are available to support a variety of speakers: the J4 terminal block, and an output audio jack (J5). A screwdriver can be used to attach the speaker(s) of choice to J4, and each speaker should have two leads: positive (red) and negative (black). The audio jack output connects both channels of the speaker to R+ and R–, so it is not true mono. The Class-D ToolStick board is intended for use with stereo systems.

The Class-D ToolStick can be powered from either the Debug (J3) or Device (J2) USB mini connectors as shown in Figure 2. The green POWER LED (DS7) will light up when the board is powered. Do not touch the Capacitive Sensing slider during power-on, as this is when the board calibrates the slider. Once the board is ready for use, the blue LED (DS1) will turn on.

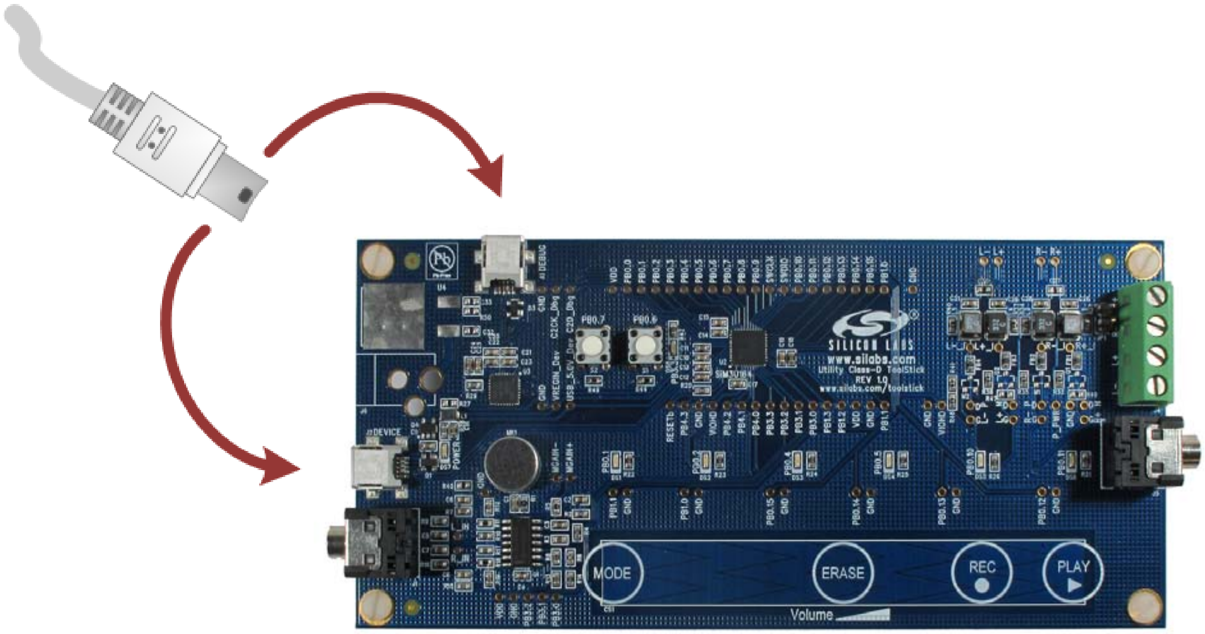


Figure 2. Powering the Board

Change the volume of the output at any time by pressing firmly and sliding on the Capacitive Sensing slider (CS1). The blue LEDs above the slider indicate the volume level.

The **MODE** button on the slider can be used to change the mode of the board:

- Mode 1 (blue LED DS1): Stereo Jack Input Mode
- Mode 2 (blue LED DS2): USB Input/Output Mode
- Mode 3 (blue LED DS3): Play Prerecorded Flash Mode
- Mode 4 (blue LED DS4): Play/Record Flash Mode

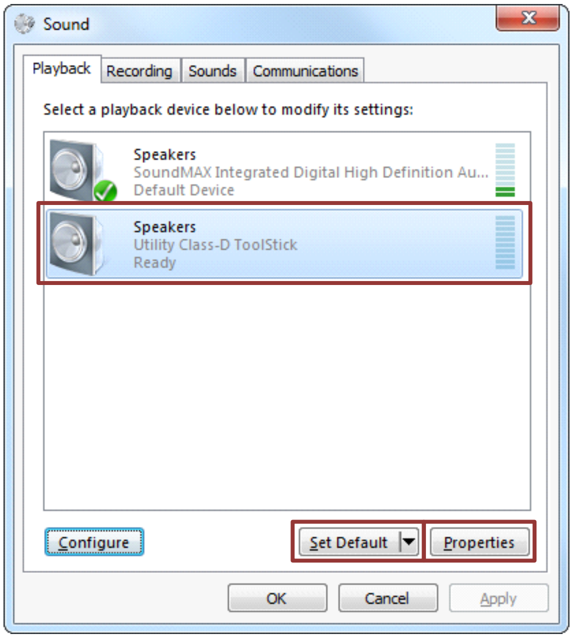
3.1. Stereo Jack Input Mode

The default mode of the Class-D ToolStick is stereo jack input mode, indicated by blue LED DS1. To play sound in this mode, connect any input source (MP3 player, phone, etc.) to the J1 stereo jack. The volume level of the input device should be as high as possible without causing saturation, which the red LED (DS8) indicates.

3.2. USB Input/Output Mode

In USB mode, the board plays sound from the USB host PC or records sound from the board microphone (MK1) and sends it up to the PC. To use the board in this mode, connect the Device USB connector (J2) to the host PC and press the **MODE** button until blue LED DS2 turns on. The board should appear as the **Utility Class-D ToolStick** in Device Manager once it's properly enumerated as a USB Audio device.

To select the Class-D Audio device as the Playback or Recording source in Windows 7, right-click on the **Volume Mixer** icon in the **System Tray** and select either **Playback devices** or **Recording devices**. In the Playback tab of the dialog, select the **Utility Class-D ToolStick** and press the **Set Default** button as shown in Figure 3. The settings for the device can be selected using the **Properties** button in this tab.



Select a playback device below to modify its settings:

Speakers
SoundMAX Integrated Digital High Definition Au...
Default Device

Speakers
Utility Class-D ToolStick
Ready

Figure 3. Configuring the USB Utility Class-D ToolStick as the Playback or Recording Device

In the Recording tab of the dialog, select the **Utility Class-D ToolStick** microphone and press the **Set Default** button. This will open an interface to the microphone and will display the recorded input volume to the right of the device name. If this volume is not registering sounds, press the **Properties** button to configure the microphone volume.

Once configured, any program on the PC should output to the Class-D ToolStick device. Ensure that the volume levels in the program and Volume Mixer are set to appropriate levels. If the device volume settings are not affecting the Class-D board, click the expansion button below the Speakers icon and select the **Speakers (Utility Class-D ToolStick)** as shown in Figure 4.

When recording, the red LED (DS8) indicates when the microphone input saturates.

Note: As long as a recording interface is open (i.e. Volume Mixer Recording devices dialog or a recording program open), the Class-D ToolStick prevents the user from switching the operation mode. This is due to potential synchronization issues when switching between different modes and sending data up to the host PC. To switch the mode, close the Volume Mixer or the recording program.

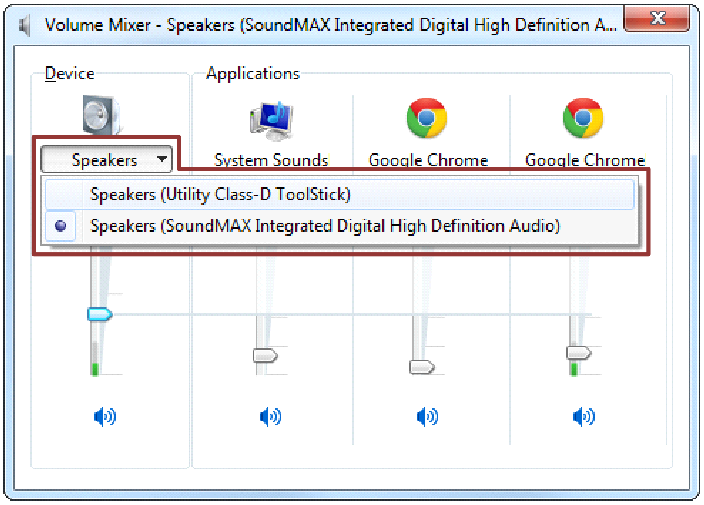

Volume Mixer - Speakers (SoundMAX Integrated Digital High Definition A...
Device
Applications
Speakers
System Sounds
Google Chrome
Google Chrome
Speakers (Utility Class-D ToolStick)
Speakers (SoundMAX Integrated Digital High Definition Audio)

Figure 4. Selecting the USB Utility Class-D ToolStick in Volume Mixer

3.3. Play Prerecorded Flash Mode

In prerecorded flash mode, the Class-D ToolStick will output ~7 seconds of sound from an array saved to flash. Enter the mode by pressing the **MODE** button until blue LED DS3 turns on.

Press the **PLAY** slider button on the board to start the waveform, and press the **PLAY** slider button again to stop it before it completes. The blue LED (DS6) will indicate when a play operation is in progress. This mode provides an alternative sound input source when no sources are available and provides an indication of what sound effects would sound like.

3.4. Play/Record Flash Mode

The play/record flash mode records compressed sound to the flash of the MCU. Enter the mode by pressing the **MODE** button until blue LED DS4 turns on.

Press the **REC** slider button to start recording, and press the **REC** slider button again to stop recording. The blue LED (DS5) will indicate when a record operation is in progress. Multiple recordings will append to the full recording in flash. Once the flash is full, the MCU will indicate no additional recordings can be made by turning on and immediately turning off the blue record LED. To record additional sound, press the **ERASE** slider button to erase the flash. The blue DS5 LED will turn on while the erase operation is in progress. When recording, the red LED (DS8) indicates when the microphone input saturates.

Press the **PLAY** slider button on the board to start the waveform, and press the **PLAY** slider button again to stop it before it completes. The blue LED (DS6) will indicate when a play operation is in progress.

4. Class-D ToolStick Hardware Overview

The Class-D ToolStick board enables Class-D application development on the SiM3U164 MCU. In addition to providing a development platform for Class-D applications, the Class-D ToolStick board serves as a low-cost, general-purpose development platform for the SiM3U1xx USB MCUs.

Figure 5 shows the Utility Class-D ToolStick board features. Full schematics for the board can be found in 7. "Schematics," on page 22

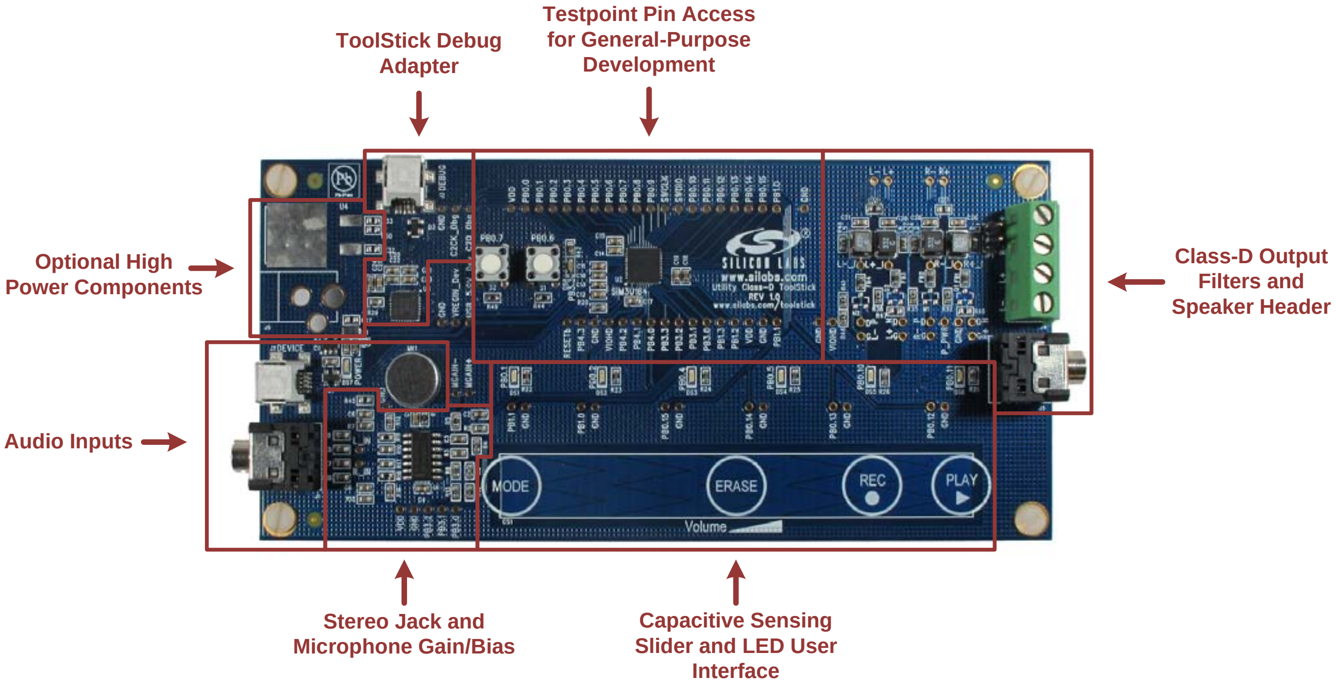

ToolStick Debug Adapter
Testpoint Pin Access for General-Purpose Development
Optional High Power Components
Class-D Output Filters and Speaker Header
Audio Inputs
Stereo Jack and Microphone Gain/Bias
Capacitive Sensing Slider and LED User Interface

Figure 5. Utility Class-D ToolStick Board Features

4.1. Push-Button Switches and LEDs (S1-2, DS1-DS8)

The Class-D ToolStick board has two push-button switches and eight LEDs summarized in Table 1. The switches connect to PB0.6 (S1) and PB0.7 (S2) and are currently unused in the Class-D firmware. The switches are normally open and pull the pin voltage to ground when pressed.

Port pins PB0.1, PB0.2, PB0.4, PB0.5, PB0.10, and PB0.11 connect to six blue LEDs (DS1-DS6). Pin PB3.3 connects to the red LED (DS8). The green LED, POWER (DS7), turns on when USB power is applied to the board from either USB connector. The LEDs connect to VIO through a current limiting resistor.

Table 1. Class-D ToolStick Switches and LEDs

GPIO Pin	Switch or LED
PB0.6	Push-Button Switch (S1)
PB0.7	Push-Button Switch (S2)
PB0.1	Blue LED (DS1)
PB0.2	Blue LED (DS2)
PB0.4	Blue LED (DS3)

Table 1. Class-D ToolStick Switches and LEDs (Continued)

GPIO Pin	Switch or LED
PB0.5	Blue LED (DS4)
PB0.10	Blue LED (DS5)
PB0.11	Blue LED (DS6)
PB3.3	Red LED (DS8)
—	Green POWER LED (DS7)

4.2. Class-D Output Network, Terminal (J4), Header (JP1), and Jack (J5)

The four high drive I/O pins on the SiM3U164 40-pin QFN package connect to the J4 terminals after going through the Class-D output network, allowing these pins to directly drive the speaker(s). The JP1 1x2 header connects to R+ and R– to interface with the kit speaker. The output jack (J5) connects R+ and R– to both channels and does not provide true mono sound. To create a mono output, connect the L_IN and R_IN test points together in the stereo jack and microphone gain/bias section of the board.

The output network consists of a ferrite bead, LC, and grounding caps for each pair of channels, R+/R– and L+/L–. These components should be tuned for the specific application speaker as discussed in 6. "Tuning the Class-D Output Network," on page 18.

Table 2. Terminal Block Pin Descriptions (J4)

Pin	I/O
1	PB4.0 / R+
2	PB4.1 / R–
3	PB4.2 / L+
4	PB4.3 / L–

4.3. ToolStick Debug Adapter (U3)

The Class-D ToolStick board features a debug adapter via the mini-B USB connector (J3) labeled **DEBUG**. This debug adapter can be used with the Precision32 (1.0.2 and higher) and ARM uVision IDEs (4.54 and higher). When using an older version of the IDEs, the debug adapter DLLs in the IDE directories must be replaced to support the ToolStick Debug Adapter. Contact technical support (“Contact Information” section) for more information.

4.4. USB Connection Options (J2)

The Class-D ToolStick board features a USB connection via the mini-B USB connector (J2) labeled **DEVICE**. This USB connector provides the USB Audio features of the board and facilitates general-purpose USB development on the AN726 device.

4.5. Capacitive Sensing Buttons and Slider (CS1)

Four capacitive sensing buttons and one slider comprised of six segments are included on the board to provide the user interface. Six GPIO pins are connected to the slider as shown in Table 3. The GPIO pins are connected to the slider in descending order from the location labeled **MODE** to the location labeled PLAY on the board.

The test points above the slider facilitate adding grounded or shrouded connectors to measure the capacitive sensing pins.

Table 3. Capacitive Sensing GPIO Connections

Location	Pin
MODE	PB1.1
—	PB1.0
ERASE	PB0.15
—	PB0.14
REC	PB0.13
PLAY	PB0.12

4.6. Analog Audio Inputs (J1 and MK1)

The stereo jack on the board (J1) allows the board to measure stereo inputs from an input source. The microphone (MK1) enables the device to measure voice data directly. These signals are conditioned by a bias and gain circuit provided by a quad op-amp IC (U1), resistors, and capacitors to a range appropriate for the SiM3U164 SARADC modules.

5. Class-D ToolStick Firmware Overview

5.1. System Overview

The system is configured for 48 MHz AHB and APB to maximize the output resolution of the EPCA.

In all modes, the EPCA0 operates continuously and outputs the PWM signal to the speaker. This waveform generation is never interrupted. The SARADC modules operate in 12-bit mode at the maximum SAR clock frequency in order to complete conversions as quickly as possible. The USB internal oscillator provides the 48 MHz system clock, which in conjunction with the PLL0 module, provides the precise timing required in the different modes to recreate the audio waveforms.

5.1.1. SiM3U164 Module Usage

The modules used by the Class-D ToolStick firmware are:

- **SARADC0**: Measures the right stereo jack channel and the microphone input
- **SARADC1**: Measures the left stereo jack channel
- **VREF0**: Provides the 2.4 V reference for the ADCs
- **FLASHCTRL0**: Writes recorded microphone data to flash
- **EPCA0**: Generates Class-D PWM waveforms (four channels)
- **CAPSENSE0**: Provides button and slider user interface
- **TIMER0**: Generates ~20 ms capacitive sensing scan timer
- **TIMER1**: Generates mode timing (e.g. 48 kHz in USB) and handles LED PWM updates
- **PCA0 and Crossbar**: Generates LED PWM waveforms for the six blue LEDs (one channel)
- **USB0**: Provides the USB Audio input/output interface
- **PLL0**: Provides the precise timing for the system and spectrum spreading to reduce emissions

AppBuilder generates the initialization code for all modules except for the USB0 module. This AppBuilder project is included in the Utility Class-D ToolStick software package.

Note: Exporting code from version 1.1.1 of AppBuilder will overwrite the project file associated with the Class-D ToolStick firmware. This project file includes manual changes to include the si32Library that will be removed when AppBuilder exports the code. A saved version of the project file (**save.project**) is available in the Class-D ToolStick firmware package in case this project file is overwritten. To restore the project file, simply copy the contents of **save.project** to the **.project** file.

5.1.2. Firmware Organization

The Class-D ToolStick code follows the AppBuilder code organization scheme. The application-specific code for each peripheral is in the **src** folder in the **my-** files, and the generated files from AppBuilder are in a **generated** folder below the **src** folder. Additional application-specific files are in the **src** folder with names associated with the functions in the file, like mulaw or LED_control.

- **class_d**: includes the main Class-D loop and any control not handled by the TIMER1L interrupt handler.
- **LED_control**: blue and red LED update functions; handles the shifting of the PCA channel on the crossbar.
- **mulaw**: implements the µLaw companding algorithms.
- **myApplication**: implements the system initialization code before passing control to class_d.
- **myBuildOptions**: required for si32Library.
- **myDataPlaneIn**: si32Library callback functions for the USB IN transactions (recording).
- **myDataPlaneOut**: si32Library callback functions for the USB OUT transactions (playing).
- **myUsbAudioDevice**: USB Audio descriptors and audio-specific si32Library function calls.
- **myUSB0**: non-audio USB si32Library functions calls for the application.
- **pre_recorded_array**: µLaw compressed prerecorded flash data.
- **volume**: handles the volume decoding and control from either the USB interface or the Capacitive Sensing slider.
- **my- files**: implement application code associated with the particular peripheral.
- **g- files**: AppBuilder generated initialization code associated with the particular peripheral.

5.2. Algorithms

The Class-D ToolStick uses two algorithms to manipulate data: Exponentially Weighted Moving Average (EWMA) and µLaw. This section also describes Remainder-Weighted Dithering as an optional algorithm.

5.2.1. Exponentially Weighted Moving Average (EWMA)

The EWMA is a form of Exponential Averaging that creates a low-pass filter. This filter performs noise shaping to help move the quantization noise out of the audible range. The Class-D ToolStick firmware uses this form of averaging to filter the ADC results.

The EWMA takes the form:

$$y(n) = \left[y(n-1) - \frac{y(n-1)}{2^m}\right] + \frac{x(n)}{2^m}$$

In this equation, y(n) is the current output, y(n-1) is the previous output, x(n) is the current input sample, and m is the weight of the average.

The weight determines how heavy the average of the EWMA is. The smaller weights (m = 1, 2) use more of the new sample and react more quickly to dramatic changes in the input. Larger weights (m = 4, 5) use more of the previous output and provide more smoothing.

The EWMA has several advantages over other forms of averaging:

- Quick to calculate (two shifts, an addition, and a subtraction)
- Only saving the previous value
- Contains history from every sample in time, which would take lots of memory in a traditional average
- The weight can be dynamically adjusted easily in firmware depending on the circumstances without dramatically changing the computational time

The EWMA is a form of IIR filter, which means it can become unstable if the cumulative error becomes too large.

5.2.2. Remainder-Weighted Dithering

The firmware quantizes the 12-bit ADC results or 16-bit signed USB data to a 9-bit EPCA PWM value. Remainder-weighted dithering is an optional algorithm that allows the Class-D ToolStick firmware to recover some of the energy lost in the LSBs during this conversion in a random, inaudible way.

This algorithm captures the LSBs of the original 12- or 16-bit data, shifts the data to the final 9-bit size, and compares against a random value of the same magnitude as the LSBs shifted away. If the original data was positive and the LSBs are greater than the random value, then the LSB of the 9-bit data is incremented by 1. Similarly, if the original data was negative and the LSBs are less than the random value, the LSB is decremented by 1. Otherwise, the data remains unchanged.

12-bit to 9-bit (3 bits):
output_value = 1931 = 0x078B
output_lsb = 0x78B & 0x7 = 3
output_value = 1931 >> 3 = 241

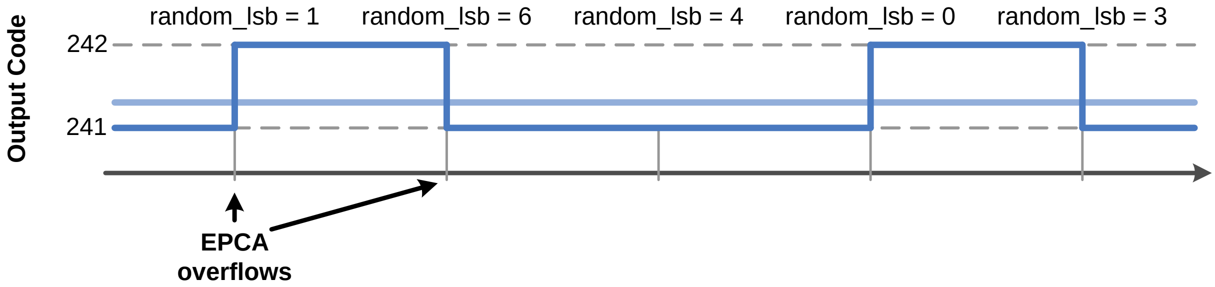


Figure 6. Remainder-Weighted Dithering Example

The example in Figure 6 assumes a original 12-bit signed data value of 1931 and shows how the average energy increases. In this particular example, the data remains the same for each EPCA overflow for simplicity, but this will not necessarily be the case in the operation of the algorithm.

5.2.3. µLaw (or muLaw)

The µLaw companding algorithm has long been used in telephony and other voice applications. This algorithm takes advantage of the behavior of the human ear by reserving most of the compression bins for low volume levels where the ear is most sensitive. Using this algorithm, the Class-D firmware can store more data in flash with no noticeable degradation in audio quality.

This algorithm takes a 14-bit signed number and adds 32 to the magnitude, which ensures that a 1 occurs in bits 5–12 of the value. This means the valid input range is –8160 to 8159. This value is then converted to an 8-bit compressed result as shown in Figure 7, where S is the sign bit. Finally, the 8-bit value is complemented.

14-bit Signed Input Data													
S	0	0	0	0	0	0	0	1	A	B	C	D	X
S	0	0	0	0	0	0	1	A	B	C	D	X	X
S	0	0	0	0	0	1	A	B	C	D	X	X	X
S	0	0	0	0	1	A	B	C	D	X	X	X	X
S	0	0	0	1	A	B	C	D	X	X	X	X	X
S	0	0	1	A	B	C	D	X	X	X	X	X	X
S	0	1	A	B	C	D	X	X	X	X	X	X	X
S	1	A	B	C	D	X	X	X	X	X	X	X	X

8-bit µLaw Encoded Data							
S	0	0	0	A	B	C	D
S	0	0	1	A	B	C	D
S	0	1	0	A	B	C	D
S	0	1	1	A	B	C	D
S	1	0	0	A	B	C	D
S	1	0	1	A	B	C	D
S	1	1	0	A	B	C	D
S	1	1	1	A	B	C	D

Figure 7. µLaw Algorithm Table

Figure 8 shows the plot of the 14-bit signed inputs versus the 8-bit compressed output.

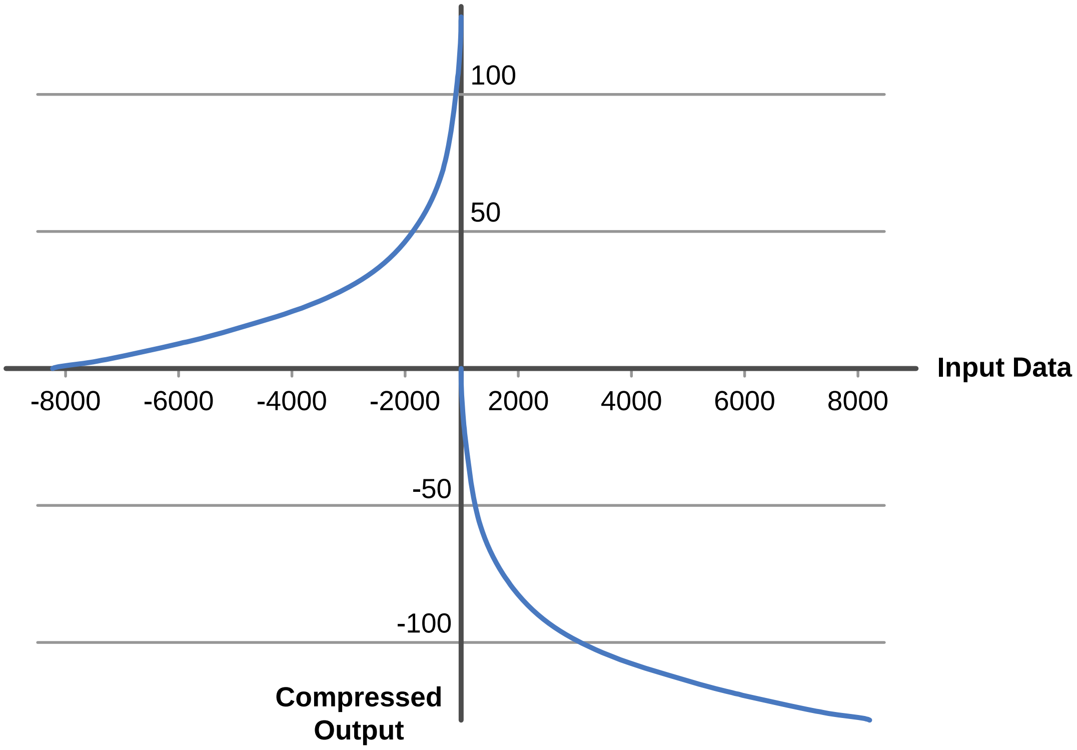


Figure 8. µLaw Algorithm Plot

5.3. EPCA0 Class-D Output

In all modes, the EPCA0 module runs in center-aligned PWM mode with an upper limit of 256. This means that the channels operate in the range of 0–511, since the channels count in half clocks. When the positive channel is on longer than the negative channel, the resulting equivalent energy is positive (above zero). When the negative channel is on longer than the positive channel, the resulting equivalent energy is negative (below zero). Combined with the components on the output, this PWM scheme generates the output waveform on the speaker as shown in Figure 9.

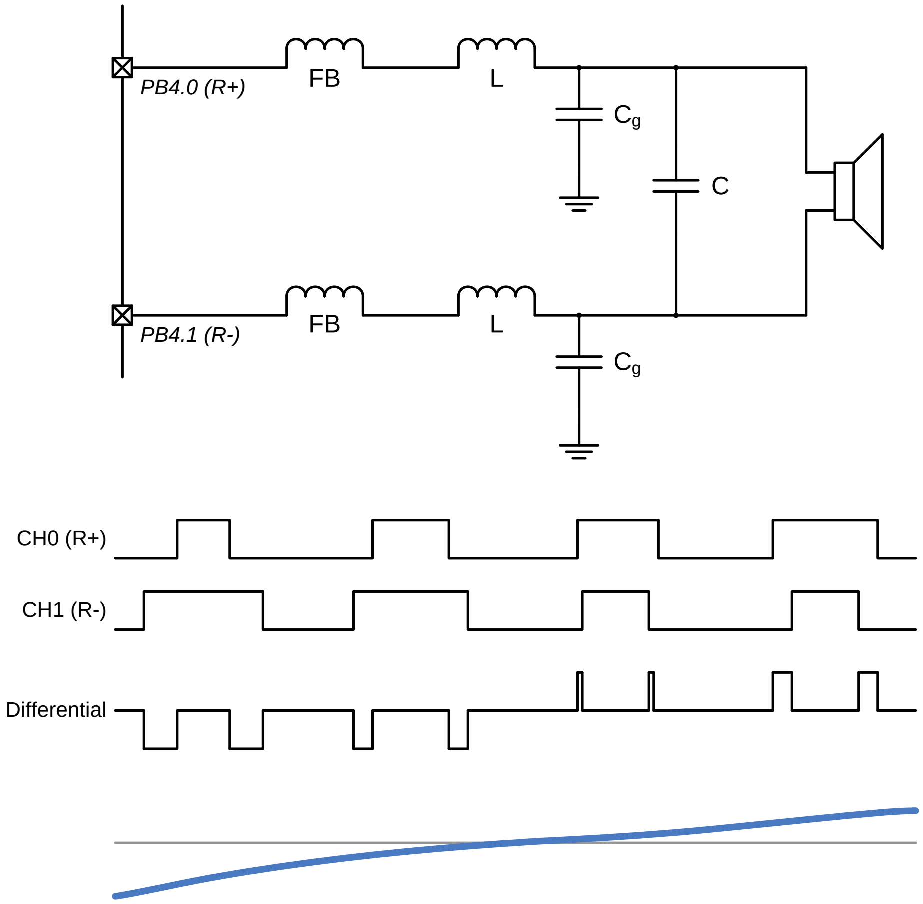


Figure 9. Class-D Output Example

Note: The output network configuration in Figure 9 is the configuration in Revision 1.0 of the hardware. An alternate output network discussed in 6. "Tuning the Class-D Output Network," on page 18 may provide better performance than the original configuration.

5.4. Volume Control

The volume control on the Class-D ToolStick board uses the current-limiting feature of the high drive I/O to adjust the output power. This has the additional benefit of allowing the PWM to always operate at full range, even at lower volumes, which reduces firmware size and improves sound quality.

5.5. Stereo Jack Input Mode

In stereo jack input mode, the goal is to sample at an even division of the EPCA0 update frequency to provide the lowest distortion attainable. The data from the ADCs is 12-bit unsigned.

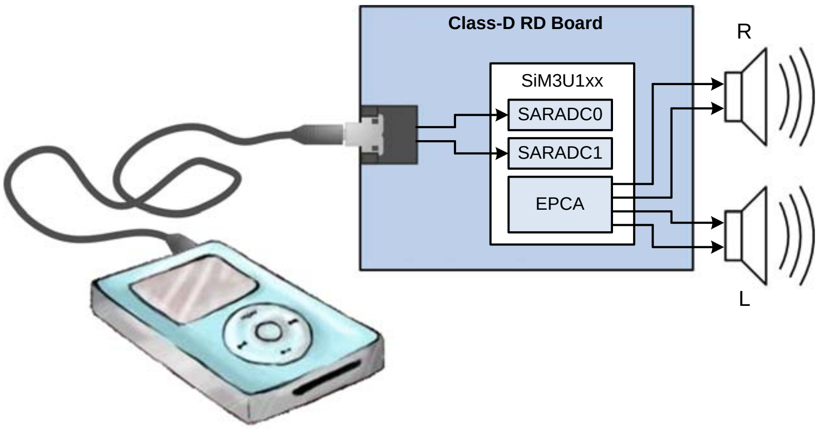


Figure 10. Stereo Jack Input Mode Block Diagram

This mode measures the input source from the stereo jack (J1) using both SARADC modules, performs the EWMA algorithm, quantizes the data to 9 bits, and outputs the PWM waveform on the EPCA0 module. Figure 11 shows the data flow for the stereo input jack mode.

The Class-D ToolStick board automatically biases the input waveform to VREF/2. On hardware that does not do this, an additional dc offset adjustment algorithm may be required.

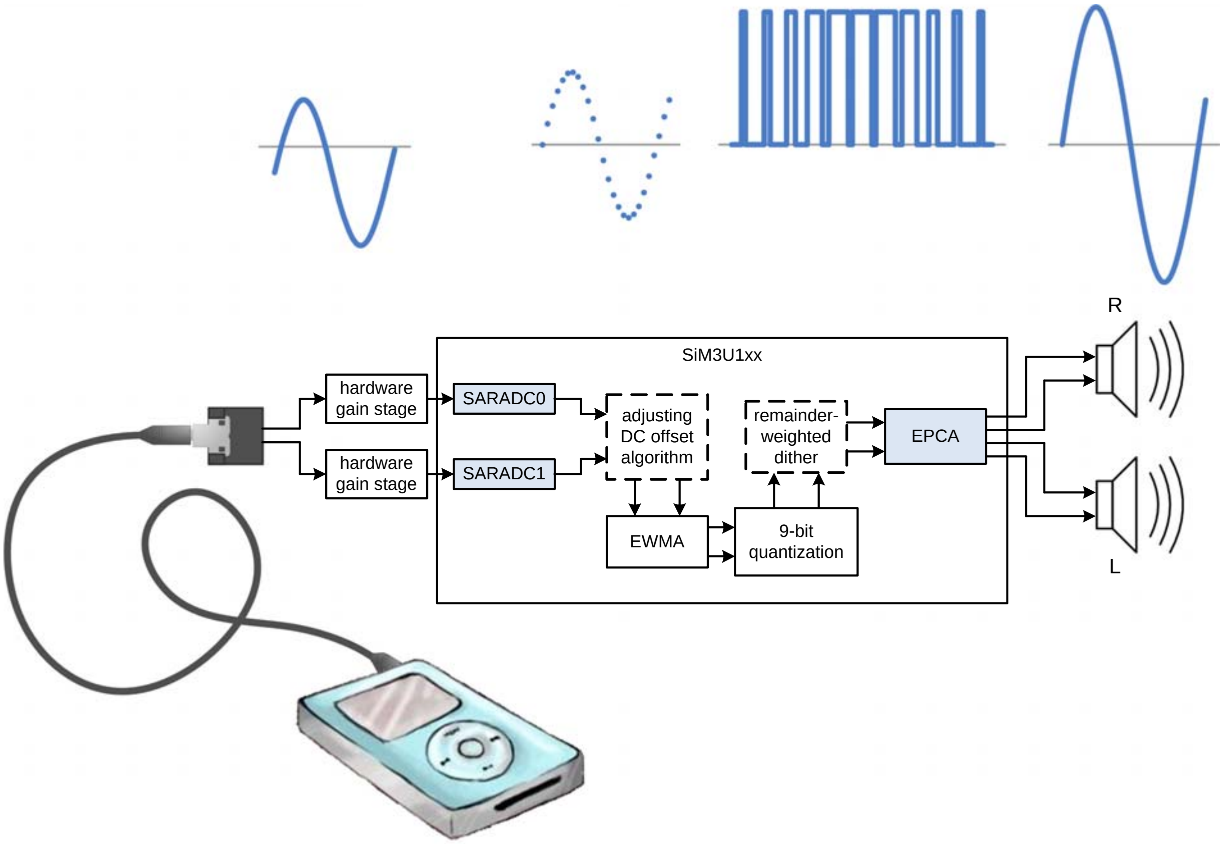


Figure 11. Stereo Jack Input Mode Flow Diagram

5.6. USB Input/Output Mode

In USB input/output mode, the system samples and outputs data at 48 kHz. The USB data is 16-bit signed data.

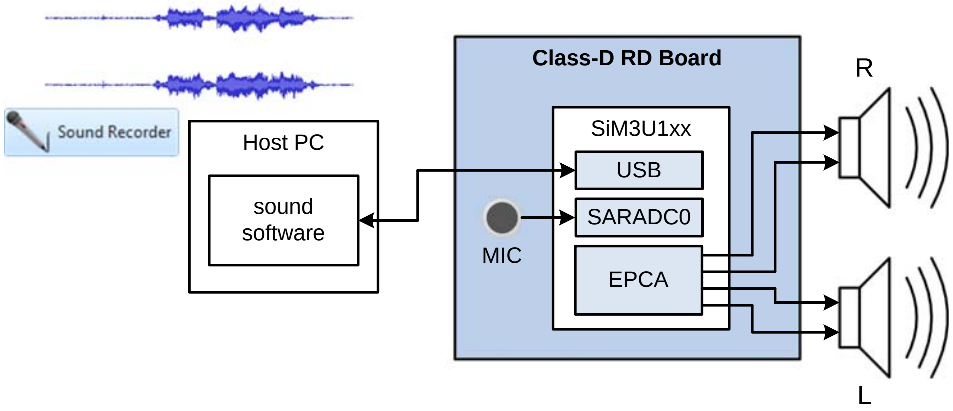


Figure 12. USB Input/Output Mode Block Diagram

For playing (from PC to the board), the board receives data from the USB interface using the si32Library USB Audio Component and outputs the PWM waveform on the EPCA0 module.

For recording (from board to the PC), the firmware measures the microphone (MK1) using one SARADC module and sends the data to the PC using the si32Library USB Audio Component.

Figure 13 shows the data flow for the USB input/output mode.

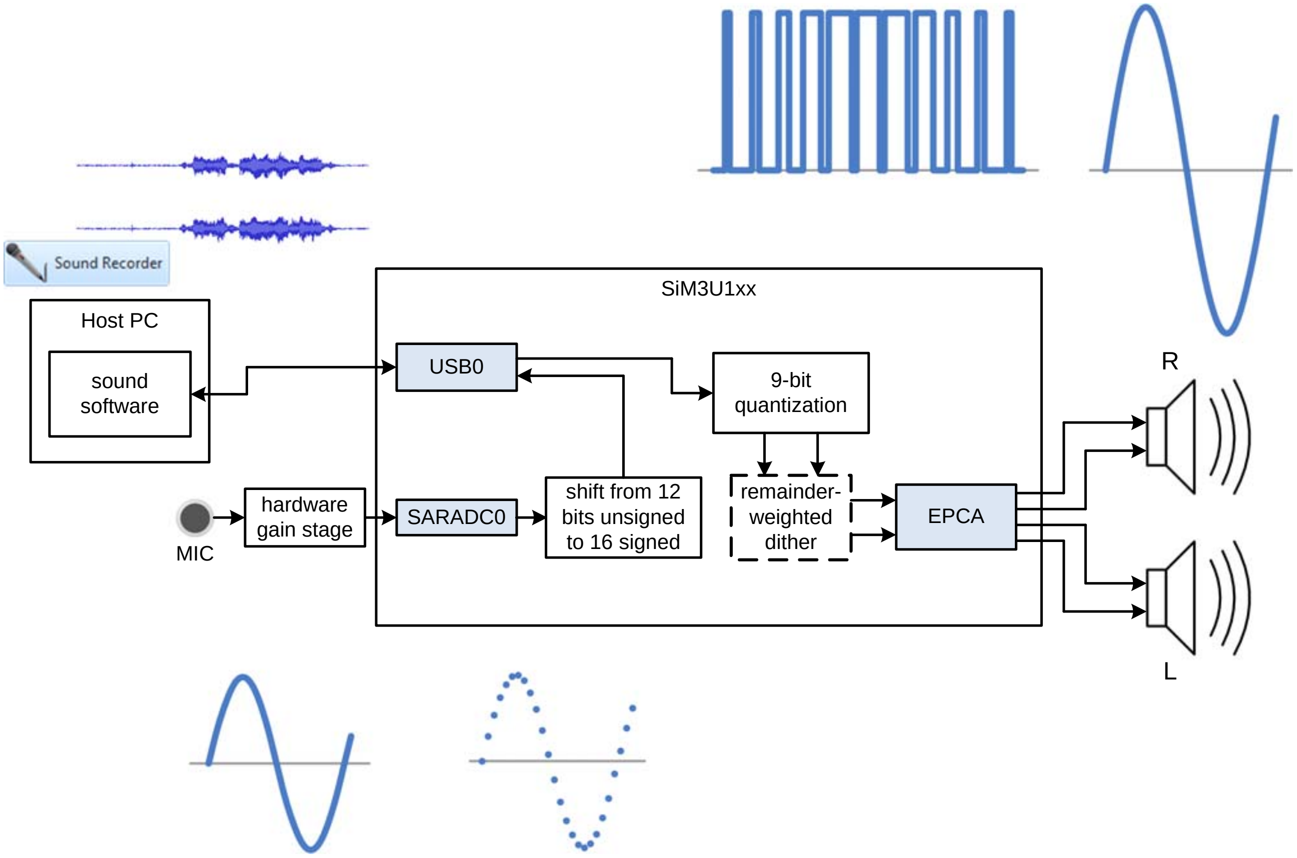


Figure 13. USB Input/Output Mode Flow Diagram

5.7. Play Prerecorded Flash Mode

In play prerecorded flash mode, the system outputs data from the flash at 9.6 kHz.

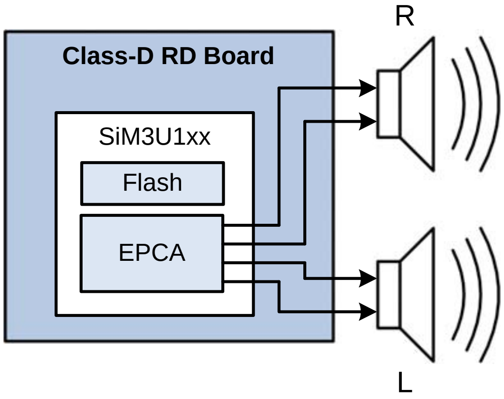


Figure 14. Play Prerecorded Flash Mode Block Diagram

The firmware reads the compressed data from the flash, decompresses the data using µLaw, performs the EWMA algorithm, quantizes the data to 9 bits, and outputs the PWM waveform on the EPCA0 module. The remainder-weighted dithering algorithm is not used at all in this mode since the data LSBs will often be 0s from the compression algorithm.

Figure 15 shows the data flow for the play prerecorded flash mode.

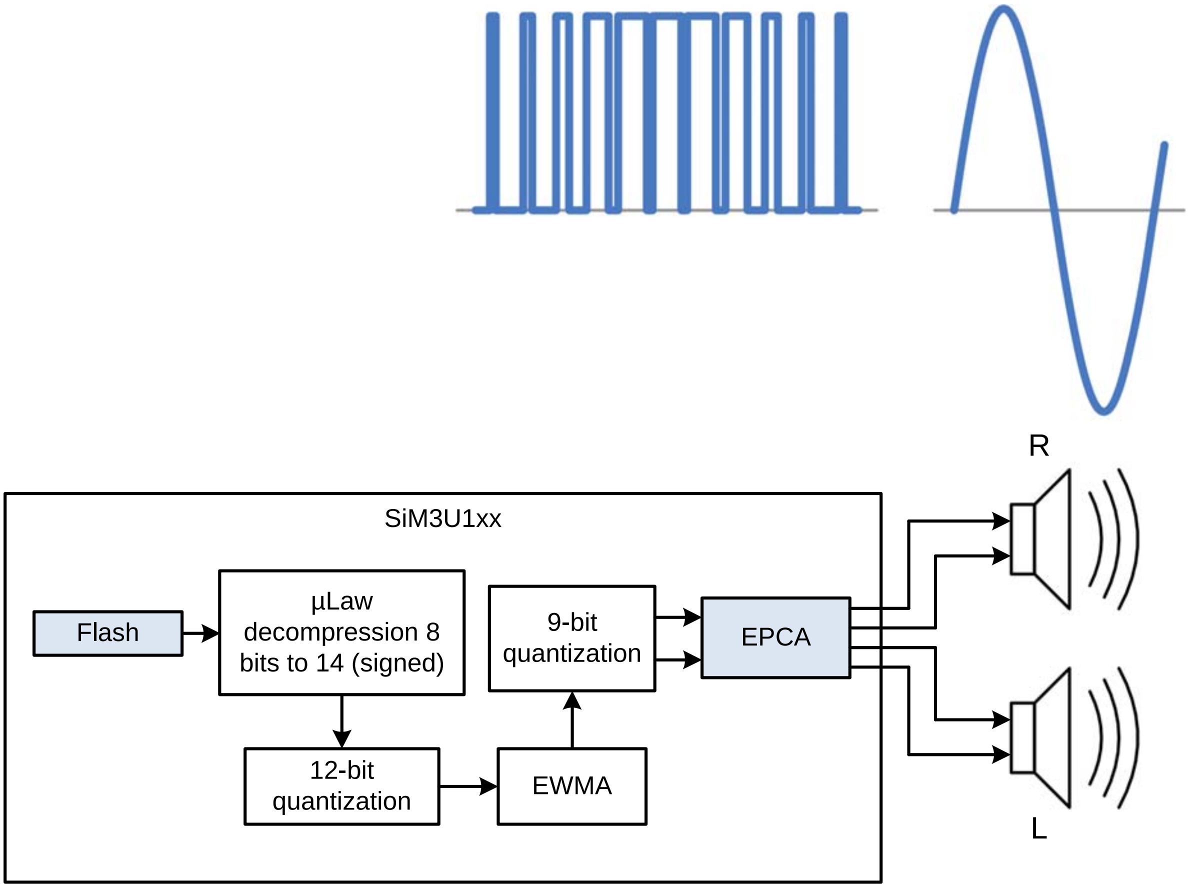


Figure 15. Play Prerecorded Flash Mode Flow Diagram

5.8. Play/Record Flash Mode

In play/record flash mode, the system samples and outputs data at 9.6 kHz.

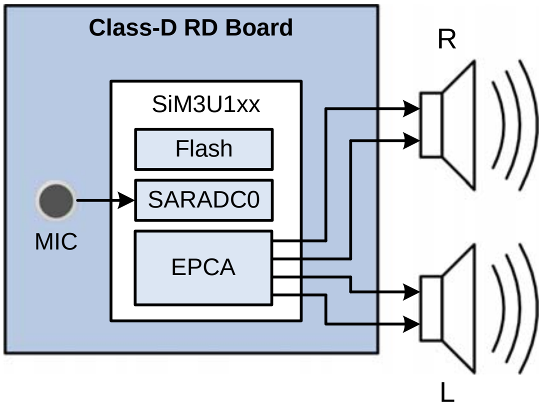


Figure 16. Play/Record Flash Mode Block Diagram

For playing, the firmware reads the compressed data from the flash, decompresses the data using µLaw, performs the EWMA algorithm, quantizes the data to 9 bits, and outputs the PWM waveform on the EPCA0 module. The remainder-weighted dithering algorithm is not used at all in this mode since the data LSBs will often be 0's from the compression algorithm.

For recording, the firmware measures the microphone (MK1) using one SARADC module, compresses the data using µLaw, and stores the data in flash.

Figure 17 shows the data flow for the play/record flash mode.

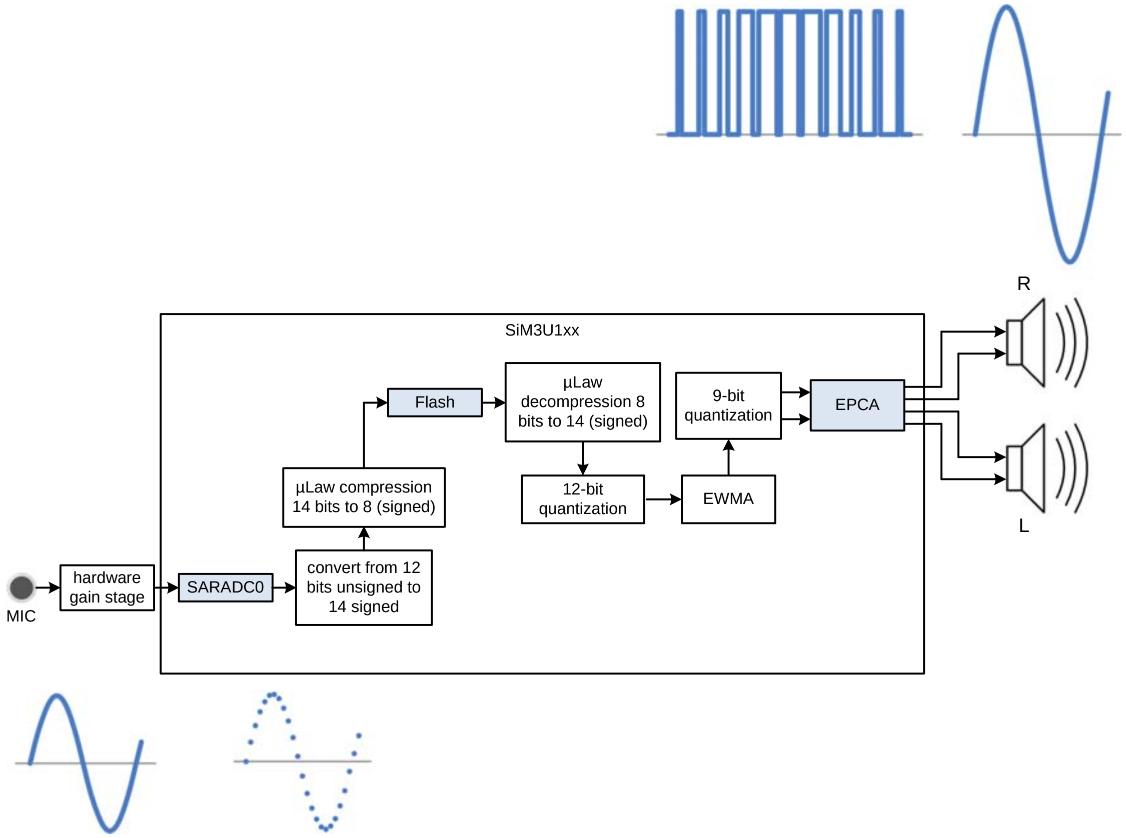

R
SiM3U1xx
Flash
µLaw decompression 8 bits to 14 (signed)
9-bit quantization
EPCA
µLaw compression 14 bits to 8 (signed)
12-bit quantization
EWMA
L
hardware gain stage
SARADC0
convert from 12 bits unsigned to 14 signed
MIC

Figure 17. Play/Record Flash Mode Flow Diagram

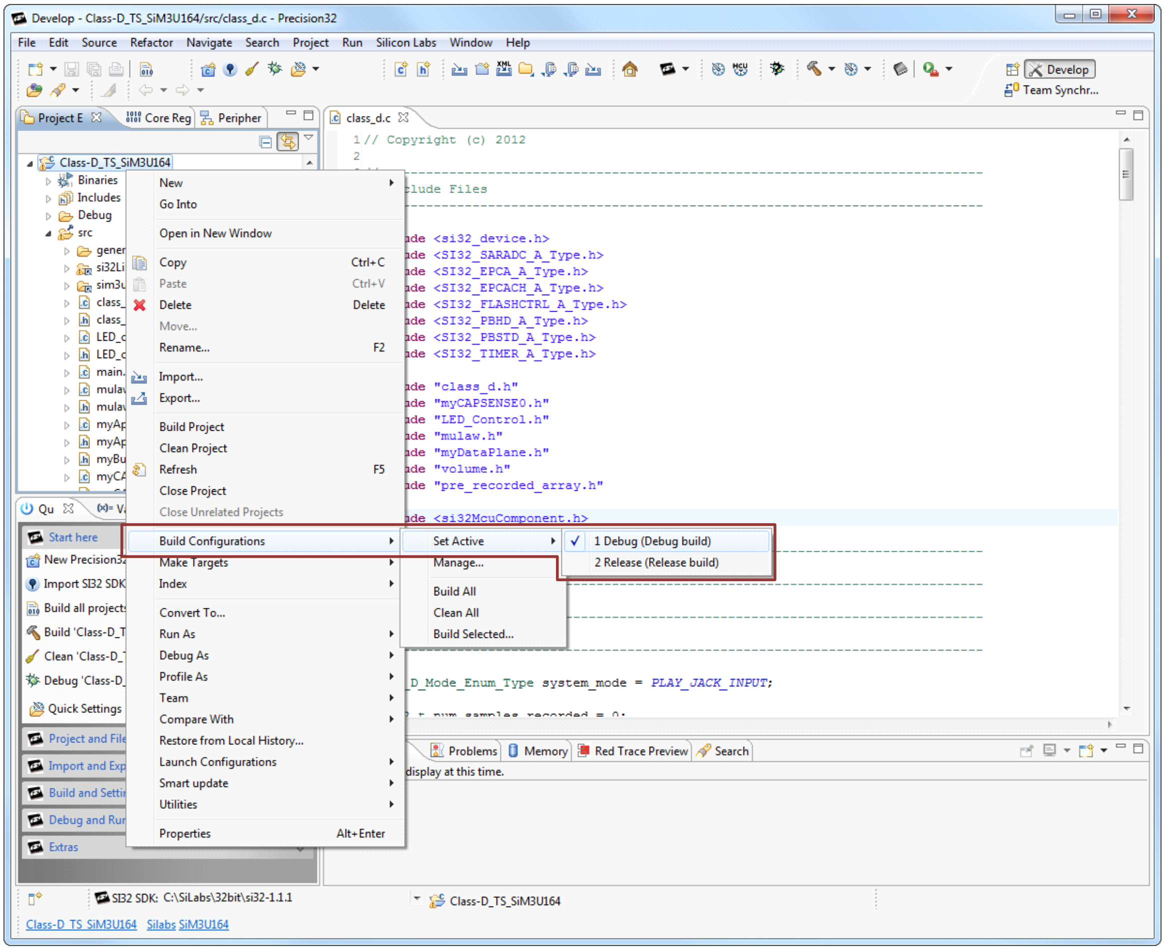
5.9. Build Configurations

The Precision32 and uVision projects include two build configurations each: Release and Debug. These configurations change the code optimization settings. In addition, each configuration sets a flag to allow the firmware to automatically detect the current build configuration and IDE. This enables the firmware to modify the play/record flash mode array size to appropriate settings based on the code size. The approximate sizes for each build configuration are shown in Table 4.

Table 4. Build Code Size Comparisons

IDE	Build	Code Size (bytes)
Precision32	Debug	68200
	Release	47200
uVision4	Debug	40200
	Release	34700

To change the active build configuration in the Precision32 IDE, right-click on the project name in the Project view and select **Build Configurations→Set Active→Debug** or **Release**.

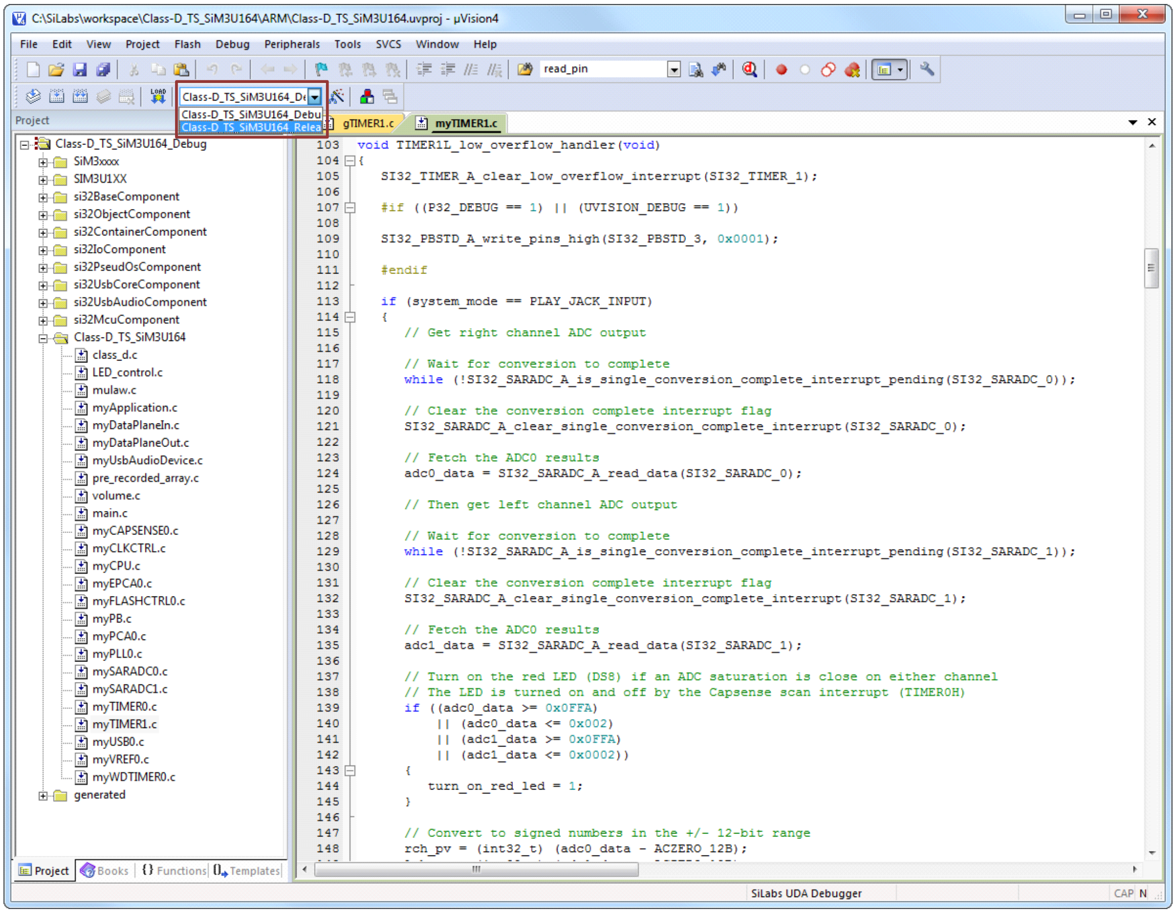


```
// Copyright (c) 2012

clude Files

de <si32_device.h>
de <SI32_SARADC_A_Type.h>
de <SI32_EPCA_A_Type.h>
de <SI32_EPCACH_A_Type.h>
de <SI32_FLASHCTRL_A_Type.h>
de <SI32_PBHD_A_Type.h>
de <SI32_PBSTD_A_Type.h>
de <SI32_TIMER_A_Type.h>

de "class_d.h"
de "myCAPSENSE0.h"
de "LED_Control.h"
de "mulaw.h"
de "myDataPlane.h"
de "volume.h"
de "pre_recorded_array.h"

de <si32McuComponent.h>

D_Mode_Enum_Type system_mode = PLAY_JACK_INPUT;
```

display at this time.

Figure 18. Setting the Active Precision32 IDE Build Configuration

To change the active build configuration in uVision4, select the configuration from the drop-down menu shown in Figure 19.



```
void TIMER1L_low_overflow_handler(void)
{
   SI32_TIMER_A_clear_low_overflow_interrupt(SI32_TIMER_1);

   #if ((P32_DEBUG == 1) || (UVISION_DEBUG == 1))

   SI32_PBSTD_A_write_pins_high(SI32_PBSTD_3, 0x0001);

   #endif

   if (system_mode == PLAY_JACK_INPUT)
   {
      // Get right channel ADC output

      // Wait for conversion to complete
      while (!SI32_SARADC_A_is_single_conversion_complete_interrupt_pending(SI32_SARADC_0));

      // Clear the conversion complete interrupt flag
      SI32_SARADC_A_clear_single_conversion_complete_interrupt(SI32_SARADC_0);

      // Fetch the ADC0 results
      adc0_data = SI32_SARADC_A_read_data(SI32_SARADC_0);

      // Then get left channel ADC output

      // Wait for conversion to complete
      while (!SI32_SARADC_A_is_single_conversion_complete_interrupt_pending(SI32_SARADC_1));

      // Clear the conversion complete interrupt flag
      SI32_SARADC_A_clear_single_conversion_complete_interrupt(SI32_SARADC_1);

      // Fetch the ADC0 results
      adc1_data = SI32_SARADC_A_read_data(SI32_SARADC_1);

      // Turn on the red LED (DS8) if an ADC saturation is close on either channel
      // The LED is turned on and off by the Capsense scan interrupt (TIMER0H)
      if ((adc0_data >= 0x0FFA)
          || (adc0_data <= 0x002)
          || (adc1_data >= 0x0FFA)
          || (adc1_data <= 0x0002))
      {
         turn_on_red_led = 1;
      }

      // Convert to signed numbers in the +/- 12-bit range
      rch_pv = (int32_t) (adc0_data - ACZERO_12B);
```

Figure 19. Setting the Active uVision4 IDE Build Configuration

5.10. si32 HAL Version

The Class-D ToolStick firmware was originally developed for v1.1.1 of the HAL or later. Compiling against previous versions of the HAL will generate errors.

6. Tuning the Class-D Output Network

6.1. Revision 1.0 Configuration

Class-D is an extension of a single-ended, single-pin speaker drive, where one side of the speaker is driven directly (perhaps with a ferrite bead) and one side is grounded. This output method works very well with SiM3U1xx devices.

The full Class-D output network consists of ferrite beads, inductor-capacitor (LC) filters, and grounding capacitors (C_g). Figure 20 shows the output network configuration used by the Revision 1.0 hardware. An alternate output network shown in Figure 21 may provide better performance than the original configuration.

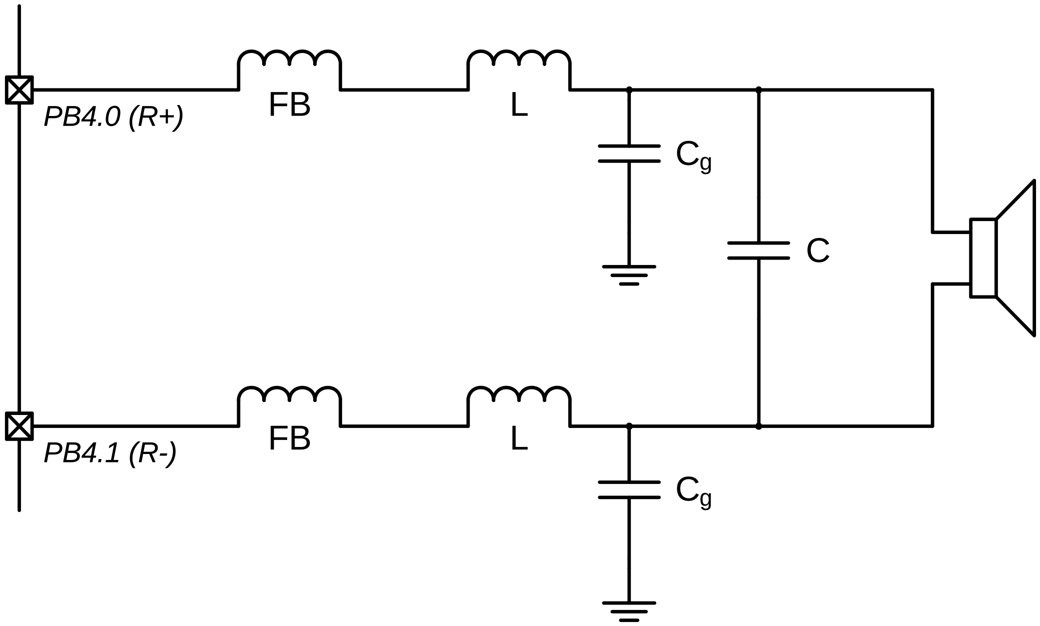


Figure 20. Revision 1.0 Hardware Class-D Output Network—Direct Drive

The ferrite beads dissipate pad switching noise as heat for EMI purposes, the LC components filter the PWM noise, and the grounding capacitors provide additional filtering in conjunction with the speaker reactance.

In almost all applications, the ferrite beads should be populated on the board. However, the other components may or may not be populated, depending on the speaker used and the cost requirements of the application. If either the L or the C are installed, both should be populated to ensure lower power consumption. In most cases, the easiest method of determining whether to install the components and the values for these components is through physical experimentation. The perceived quality can be determined by the following characteristics:

- loudness
- changes in loudness across bandwidth
- bandwidth
- noise harmonics across bandwidth
- hiss at low and high volumes

Other factors to consider are the application’s sound source (i.e. USB data versus sampled ADC data) and the maximum power consumption of the board in the application, which will determine how much data processing can occur and if external MOSFETs are appropriate. The Class-D board provides 0 Ω resistor pads to bypass the series components, if desired.

6.2. Alternate Configuration

Figure 20 shows the output network configuration used by the Revision 1.0 hardware. An alternate output network shown in Figure 21 may provide better performance than the original configuration.

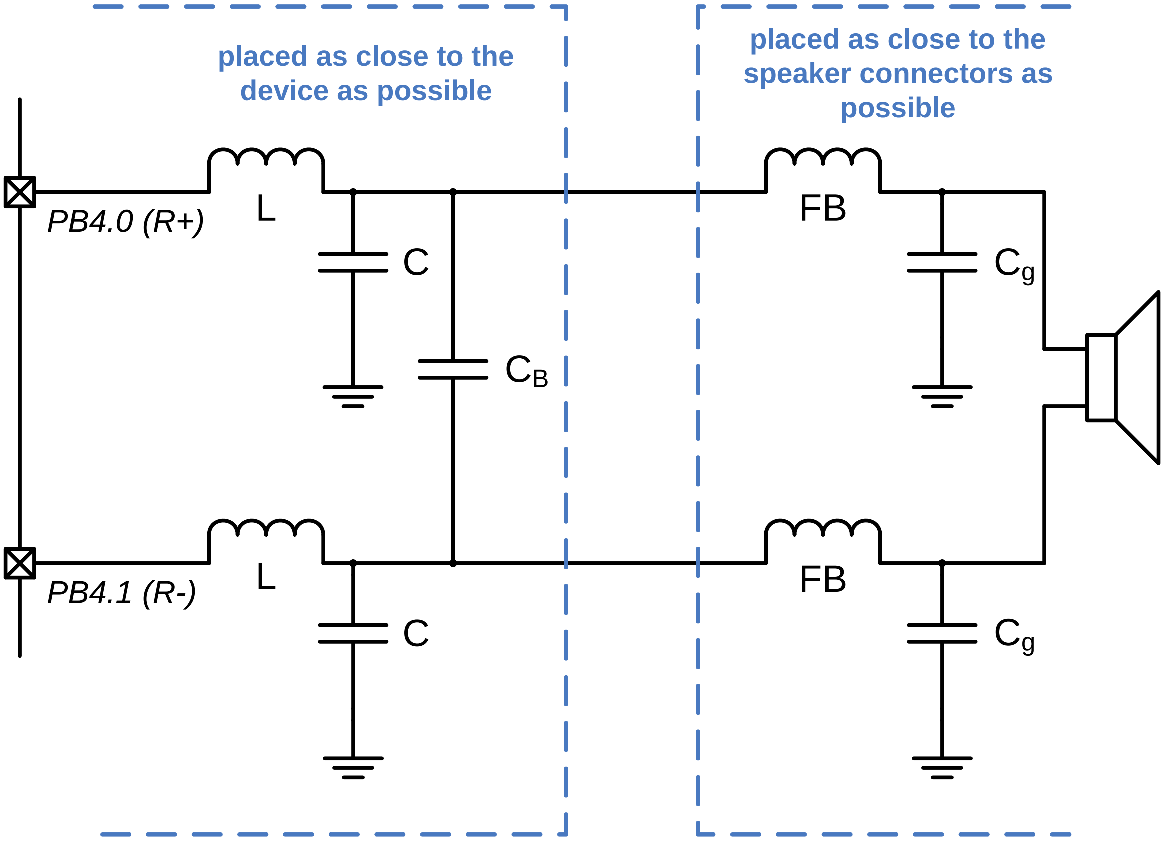

placed as close to the device as possible
placed as close to the speaker connectors as possible
PB4.0 (R+)
L
C
C_B
FB
C_g
PB4.1 (R-)
L
C
FB
C_g

Figure 21. Alternate Class-D Output Network—Direct Drive

For the alternate configuration, the recommended components for the board are shown in Table 5.

Table 5. Class-D ToolStick Alternate Output Network Recommended Components

Reference	Value
L	~3.3 µH, 1.5 A
C	~2.2 µF
C_g	~20 pF
C_B	~0.47 µF

6.3. Revision 1.0 High Power Configuration

Instead of directly driving the speaker, the high-drive I/O can interface with external MOSFETs as shown in Figure 22, enabling higher-power applications. The board includes footprints for these external MOSFETs to enable development with this configuration. The pull-up and pull-down resistors (~100 kΩ) ensure that the tristate reset state of the high drive I/O does not cause the FETs to accidentally conduct current. When using these external MOSFETs, uninstall the R32, R35, R38, and R41 0 Ω resistors.

Figure 23 illustrates an alternate full-bridge high-power output network. The diode and RC circuits on the MOSFET gates ensure that the two devices will not both be conducting simultaneously.

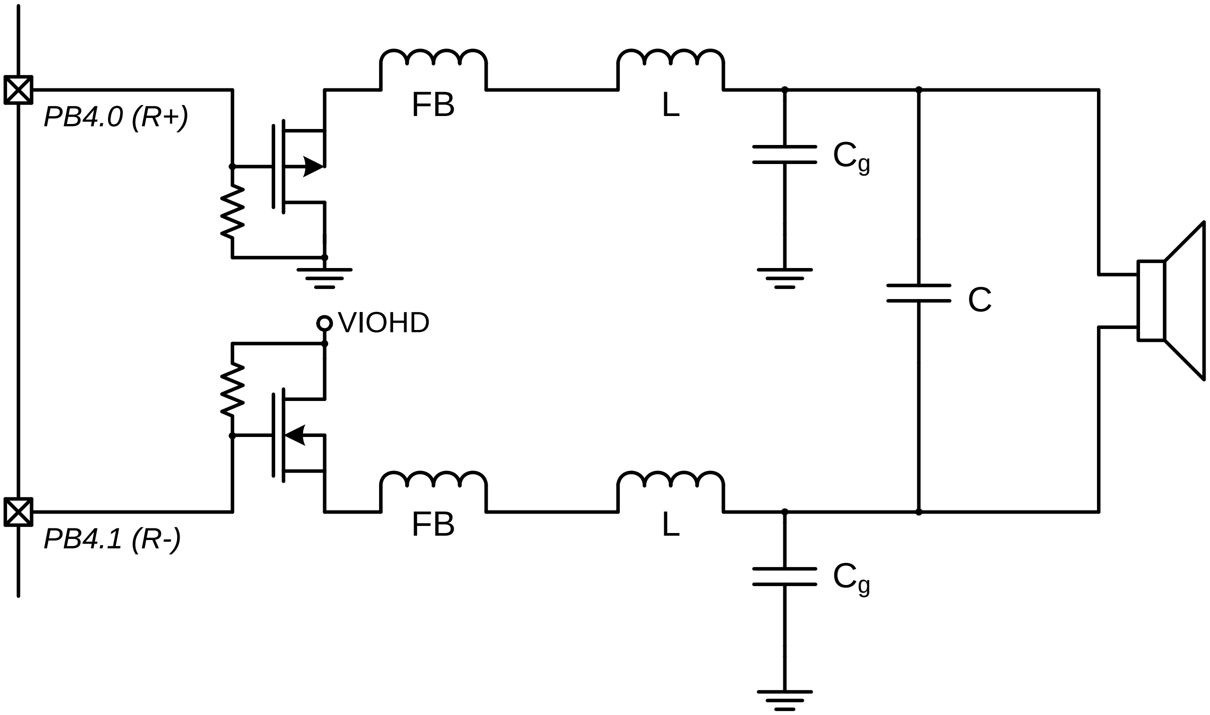


Figure 22. Half-Bridge Class-D Output Network–External MOSFETs

The Revision 1.0 hardware provides footprints to a 9 V dc adapter connector and high power LDO. These components can be populated to provide a higher current source to the external MOSFET transistors and VIOHD. Alternatively, an external supply can connect to the transistors alone by removing the R48 0 Ω resistor and placing the positive terminal on the **P_PWR** test point.

6.4. Alternate High Power Configuration

The Revision 1.0 hardware uses a half-bridge configuration shown in Figure 22. This configuration can lead to higher distortion. Figure 23 illustrates an alternate full-bridge high-power output network. The diode and RC circuits on the MOSFET gates ensure that the two devices will not both be conducting simultaneously. The value of these components should be tuned for the gate capacitance of the FETs.

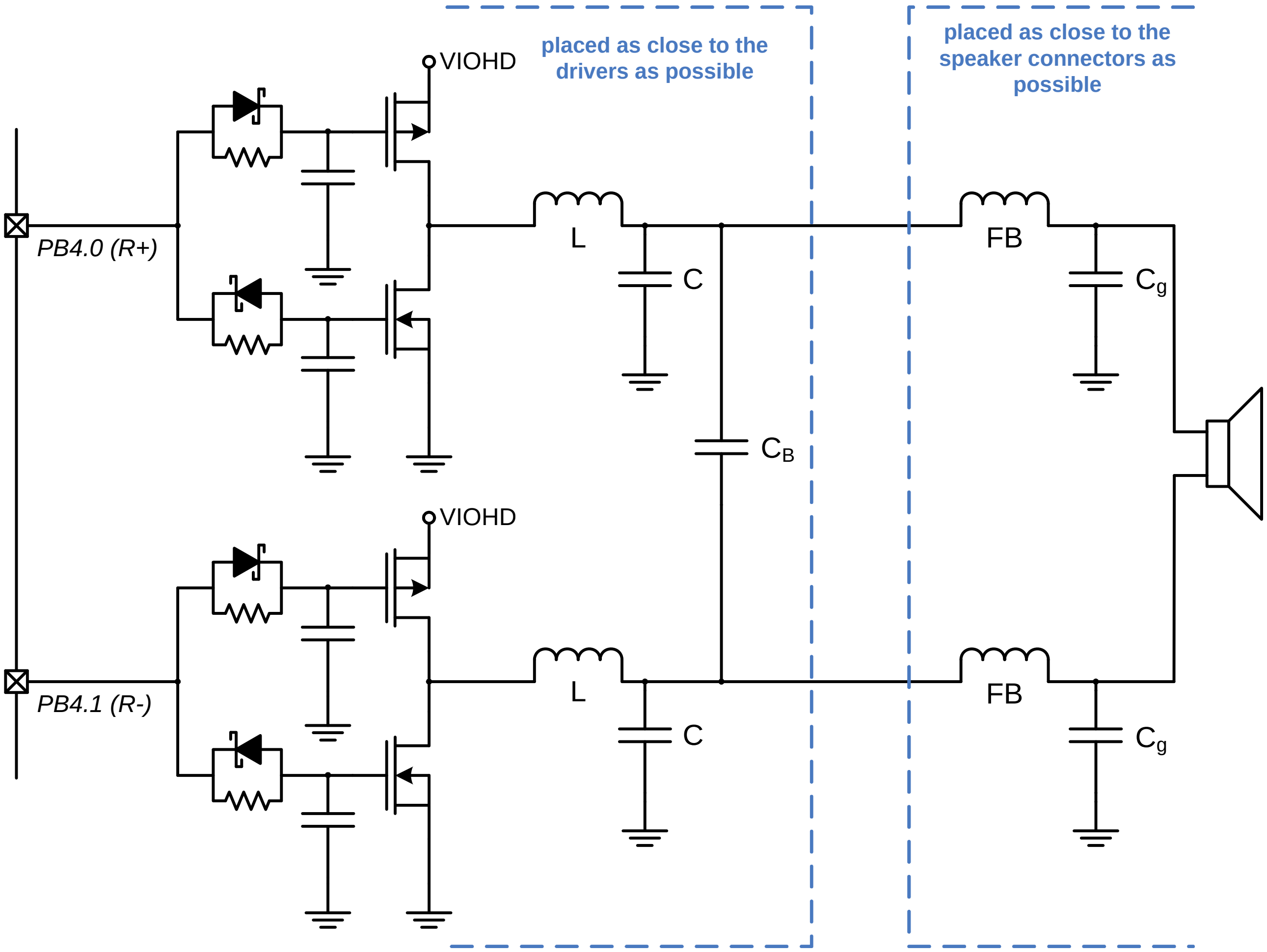

placed as close to the drivers as possible
placed as close to the speaker connectors as possible
VIOHD
VIOHD
PB4.0 (R+)
PB4.1 (R-)
L
C
L
C
C_B
FB
C_g
FB
C_g

Figure 23. Full-Bridge Class-D Output Network—External MOSFETs

7. Schematics

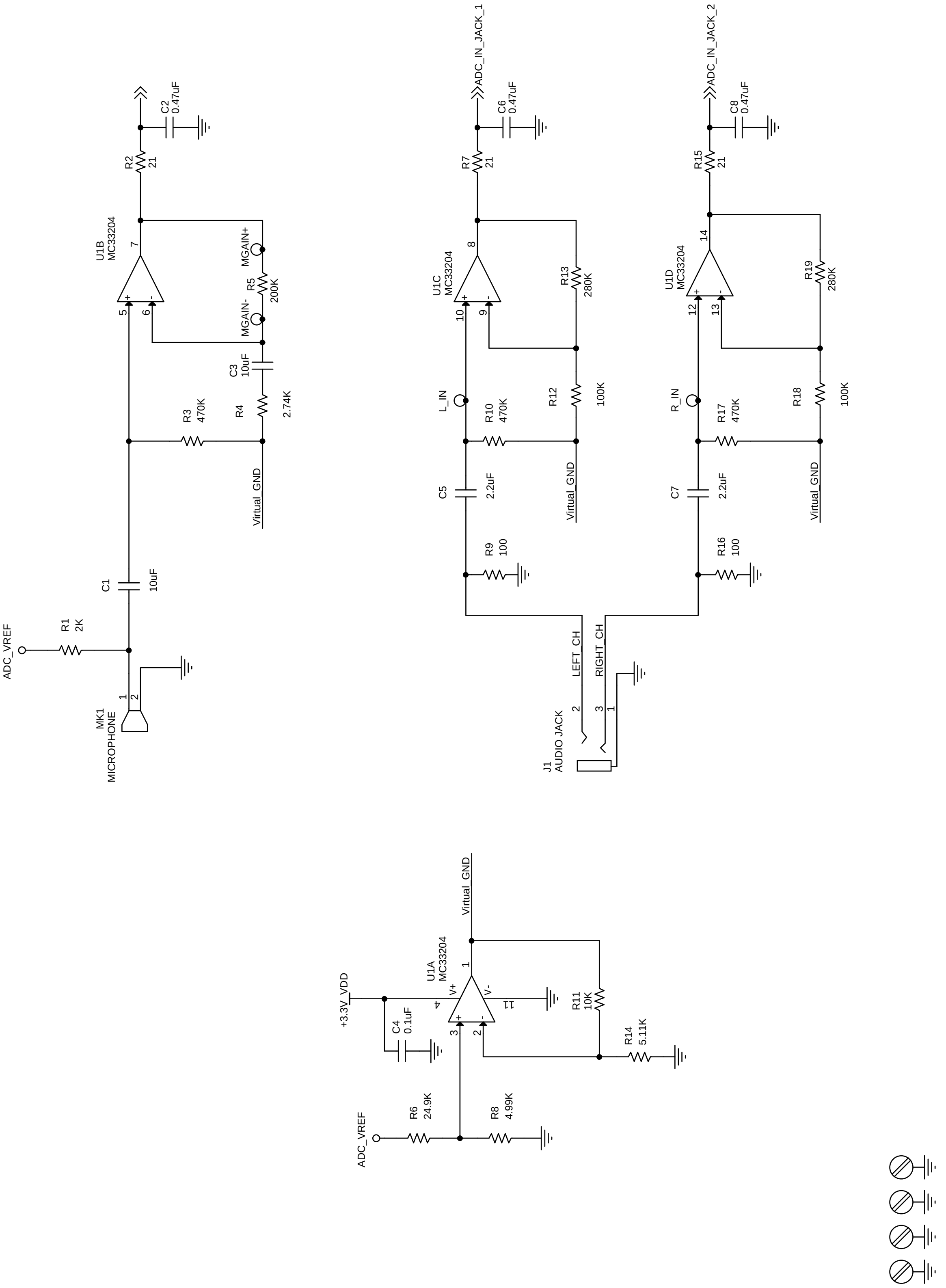


Figure 24. Utility Class-D ToolStick Board Schematic (1 of 3)—Revision 1.0

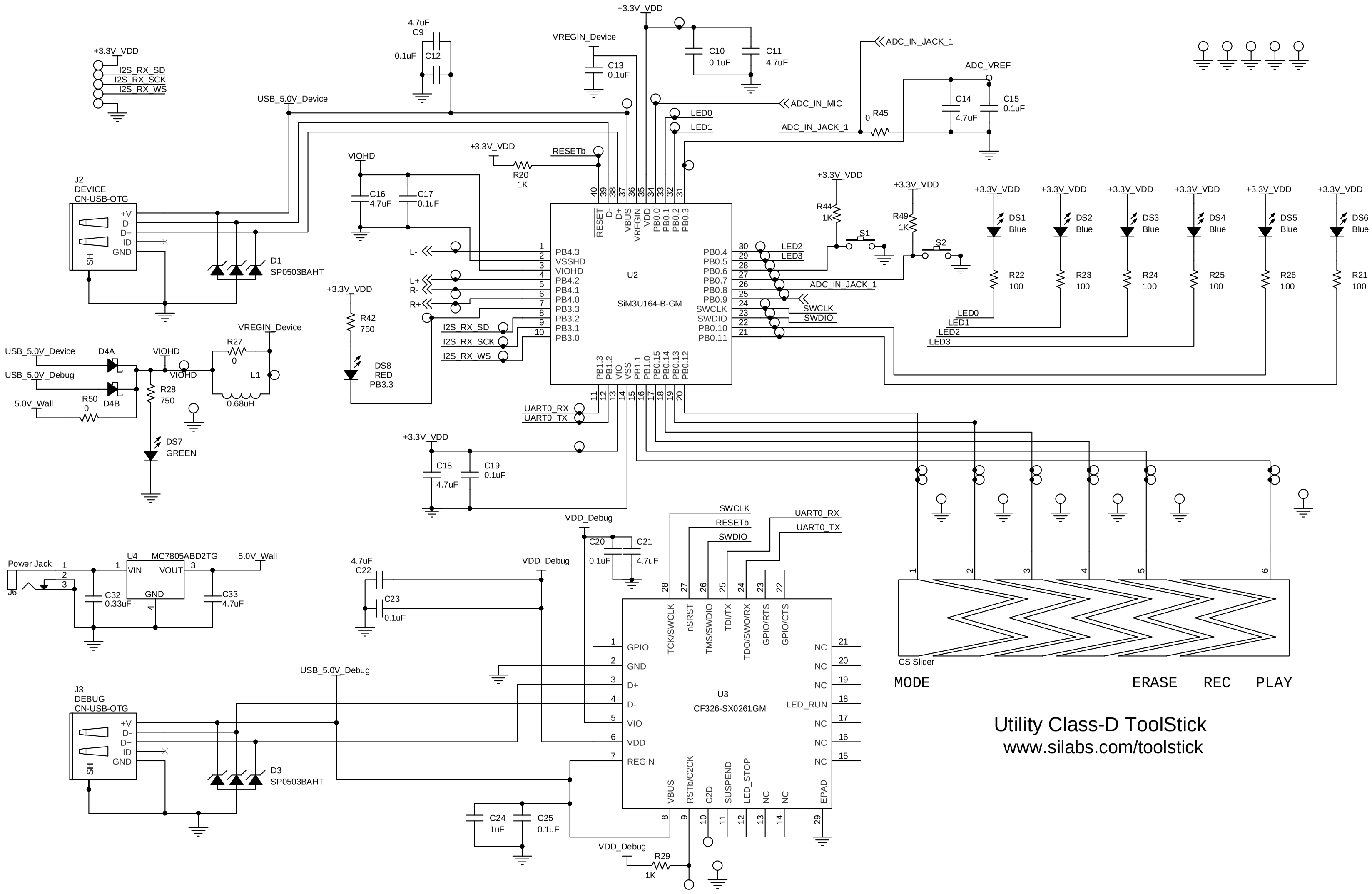


Figure 25. Utility Class-D ToolStick Board Schematic (2 of 3)—Revision 1.0

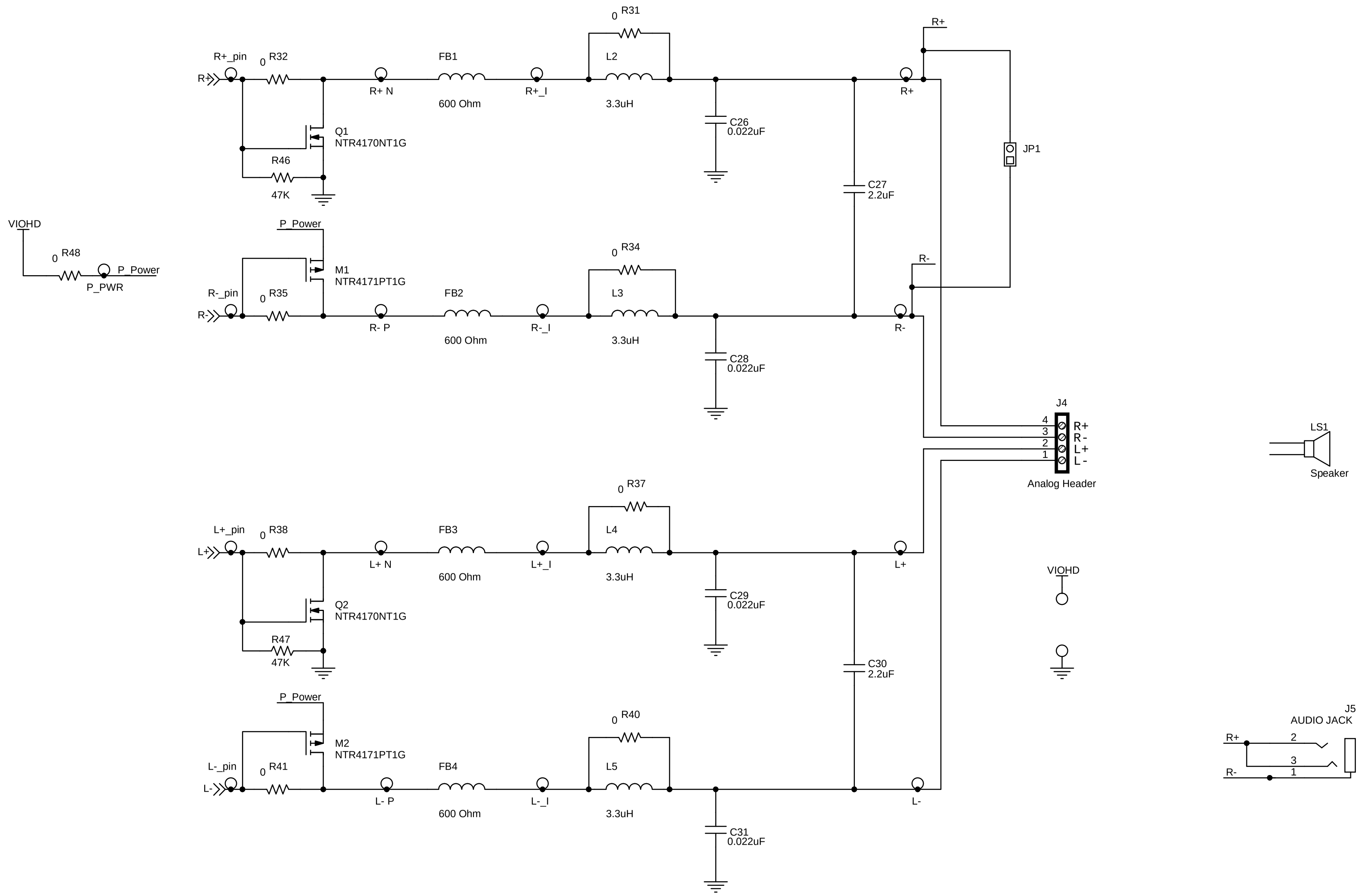


Figure 26. Utility Class-D ToolStick Board Schematic (3 of 3)—Revision 1.0

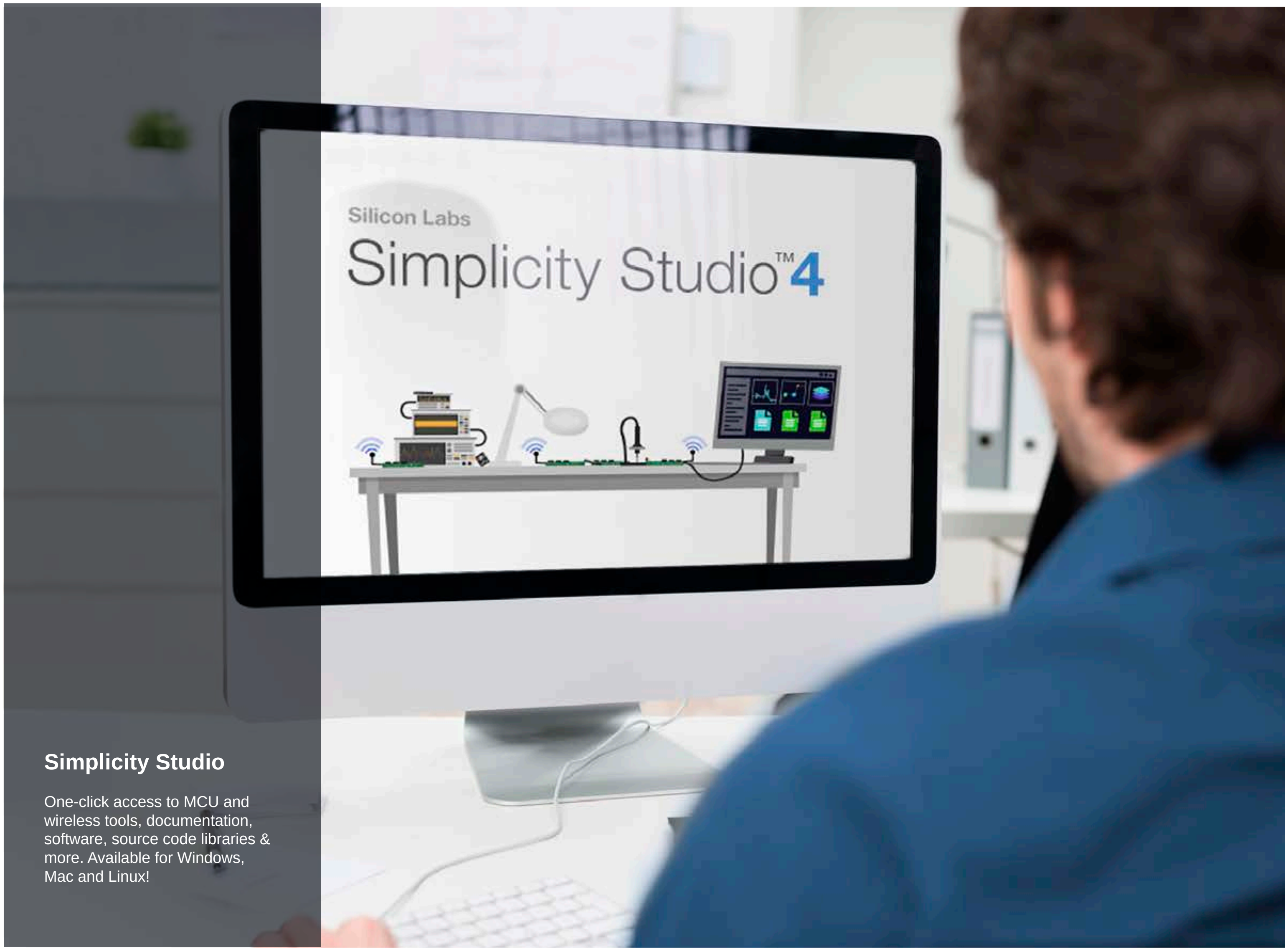
8. Bill of Materials

Table 6. Utility Class-D ToolStick Board Bill of Materials

Reference	Part Number	Source	Description
C1, C3	C0603X5R6R3-106M	Venkel	10 µF, 6.3 V ±r% X5R 0603
C2, C6, C8	C0603X7R250-474K	Venkel	0.47 µF, 25 V ±10% X7R 0603
C24	C0603X7R100-105K	Venkel	1 µF, 10 V ±10% X7R 0603
C26, C28, C29, C31	C1608X7R1H223M	TDK Corporation	0.022 µF, 50 V ±20% X7R 0603
C4, C10, C12, C13, C15, C17, C19, C20, C23, C25	C0603X7R100-104K	Venkel	0.1 µF, 10 V ±10% X7R 0603
C5, C7	C0603X7R100-225K	Venkel	2.2 µF, 10 V ±10% X7R 0603
C9, C11, C14, C16, C18, C21, C22	C0603X5R6R3-475K	Venkel	4.7 µF, 6.3 V ±10% X5R 0603
D1, D3	SP0503BAHTG	Littlefuse	USB Diodes 300 mW, 20 V TVS SOT143
D4	RB550EATR	Rohm Semiconductor	Diode 700 mA, 30 V Dual Schottky SOT5N
DS1, DS2, DS3, DS4, DS5, DS6	LB Q39G-L2N2-35-1	OSRAM Opto Semiconductors Inc	Blue 15 mA, 2.85 V SMT, ChipLED 0603
DS7	SML-LX0603SUGW	LUMEX INC	GREEN 25 mA 0603
DS8	SML-LX0603IW	LUMEX INC	RED 30 mA 0603
FB1, FB2, FB3, FB4	MPZ1608S601A	TDK Corporation	600 Ω, 1 A SMT 0603
J1, J5	SJ1-3543N	CUI Inc	Stereo Audio Jack
J2, J3	UX60-MB-5ST	HIROSE	USB mini connector B
J4	1729144	Phoenix Contact	1x4 Terminal Block
JP1	TSW-102-07-T-S	Samtec	1x2 Jumper Header
L1	NLV25T-R68J-PF	TDK	0.68 µH, 300 mA ±5% GP
L2, L3, L4, L5	84332C	Murata	3.3 µH, 1.5 A ±20% Unshielded
LS1	GA0571H	ALC	Speaker 1 W Max
MK1	CMA-4544PF-W	CUI Inc	MICROPHONE MIC-2.5X9.7MM-RAD
R1	CR0603-10W-2001F	Venkel	2 kΩ, 1/10 W ±1% ThickFilm 0603
R11, R18	CR0603-10W-1003F	Venkel	100 kΩ, 1/10 W ±1% ThickFilm 0603
R12, R19	CR0603-16W-2803F	Venkel	280 kΩ, 1/16 W ±1% ThickFilm 0603
R13	CR0603-10W-1002F	Venkel	10 kΩ, 1/10 W ±1% ThickFilm 0603
R14	CR0603-8W-5111F	Venkel	5.11 kΩ, 1/8 W ±1% ThickFilm 0603
R2, R7, R15	CR0603-16W-21R0F	Venkel	21 Ω, 1/16 W ±1% ThickFilm 0603
R20, R29, R44, R49	CR0603-10W-1001F	Venkel	1 kΩ, 1/10 W ±1% ThickFilm 0603
R28, R42	CR0603-16W-7500F	Venkel	750 Ω, 1/10 W ±1% ThickFilm 0603

Table 6. Utility Class-D ToolStick Board Bill of Materials (Continued)

Reference	Part Number	Source	Description
R3, R10, R17	CR0603-10W-4703F	Venkel	470 kΩ, 1/10 W ±1% ThickFilm 0603
R31, R32, R34, R35, R37, R38, R40, R41, R48	CR0603-16W-000	Venkel	0 Ω, 1 A ThickFilm 0603
R4	CR0603-16W-2741F	Venkel	2.74 kΩ, 1/16 W ±1% ThickFilm 0603
R5	CR0603-10W-2003F	Venkel	200 kΩ, 1/10 W ±1% ThickFilm 0603
R6	CR0603-10W-2492F	Venkel	24.9 kΩ, 1/10 W ±1% ThickFilm 0603
R8	CR0603-16W-4991F	Venkel	4.99 kΩ, 1/16 W ±1% ThickFilm 0603
R9, R16, R21, R22, R23, R24, R25, R26	CR0603-10W-1000F	Venkel	100 Ω, 1/10 W ±1% ThickFilm 0603
S1, S2	EVQ-PAD04M	Panasonic Corp	Momentary Tactile Switch 6.5x4.5
SC1, SC2, SC3, SC4	NSS-4-4-01	Richco Plastic Co	4-40 HDW
SO1, SO2, SO3, SO4	2397	SPC Technology	STANDOFF HDW
U1	MC33204DR2	ON Semiconductor	Quad Op-Amp
U2	SiM3U164-B-GM	Silicon Laboratories	SiM3U164-B-GQ MCU QFN40 M6x6 P0.5
U3	CF326-SX0261GM	Silicon Laboratories	ToolStick Debug Adapter MCU
Components Not Installed			
C27, C30	C0603X7R100-225K	Venkel	2.2 µF, 10 V ±10% X7R 0603 Not Installed (NI)
C32	C0603X7R100-334K	Venkel	0.33 µF, 10 V ±10% X7R 0603 Not Installed (NI)
C33	C0603X5R6R3-475K	Venkel	4.7 µF, 6.3 V ±10% X7R 0603 Not Installed (NI)
J6	RAPC722X	Switchcraft Inc.	3-pin Power Jack 5A BARREL Not Installed (NI)
M1, M2	NTR4171PT1G	ON Semiconductor	–3.5 A, –30 V Pchan SOT23 Not Installed (NI)
Q1, Q2	NTR4170NT1G	ON Semiconductor	3.2 A, 30 V Nchan SOT23 Not Installed (NI)
R27, R45, R50	CR0603-16W-000	Venkel	0 Ω, 1 A ThickFilm 0603 Not Installed (NI)
R46, R47	CR0603-16W-4702F	Venkel	47 kΩ, 1/16 W ±1% ThickFilm 0603 Not Installed (NI)
U4	MC7805ABD2TG	ON Semiconductor	1 A max LDO D2PAK Not Installed (NI)



Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support and Community
community.silabs.com

Disclaimer
Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Labs shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

Trademark Information
Silicon Laboratories Inc.® , Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world’s most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISOmodem®, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, Telegesis, the Telegesis Logo®, USBXpress® and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.

SILICON LABS

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

http://www.silabs.com