USB Headphone Amplifier

Masaya Honda

jp.masaya@gmail.com

Summary

For this project, I made my own USB-powered headphone amplifier that is capable of driving 300Ω headphones. The idea stems from the consistent hissing noise and poor sound quality when using my laptop’s built-in headphone jack. I performed calculations by analyzing component datasheets to ensure that the components that were chosen meet the specifications that would allow for it to drive the headphones under bus-powered operation. I drew the schematic using Altium after deciding on what components will be used in the design. Then, I worked on the PCB layout for manual routing and used techniques to minimize noise and interference. I placed the decoupling capacitors as close to the pins of the ICs as possible, used polygon pours to create low impedance return paths around sensitive components. To avoid any interference between analog/digital signals, analog, and digital-based components were separated.

Upon soldering all of the components and finishing the assembly of the headphone amplifier, I was faced with multiple issues that prevented the device from working. One main issue was based on how the USB bridge IC was configured. I overlooked the simple mistake of not configuring all of the pins necessary to run the IC in bus-powered mode. This was eventually fixed by altering the connections made on the PCB with wires and made an update to the PCB file for future implementations. Despite these mistakes, I was able to get the headphone amplifier working as intended. I am very pleased with how the headphone amplifier turned out in terms of its functionality, sound quality, and aesthetics of the headphone amplifier.

Table of Contents

1. Introduction

The project idea was inspired by my laptop's subpar sound quality combined with its hissing noises when using its headphone jack. To avoid using the built-in headphone jack, I wanted to design a good-sounding USB headphone amplifier that is able to drive over-the-ear headphones (300Ω) without an external power supply. The process of making this a reality included component selection to meet specifications, analyzing datasheets, performing schematic drawing and PCB layout in Altium, assembly, and troubleshooting. The two main ICs that were used in my design include the PCM2706C USB bridge IC and the PCM5102A DAC IC from Texas Instruments.

2. Design Choices and Component Selection

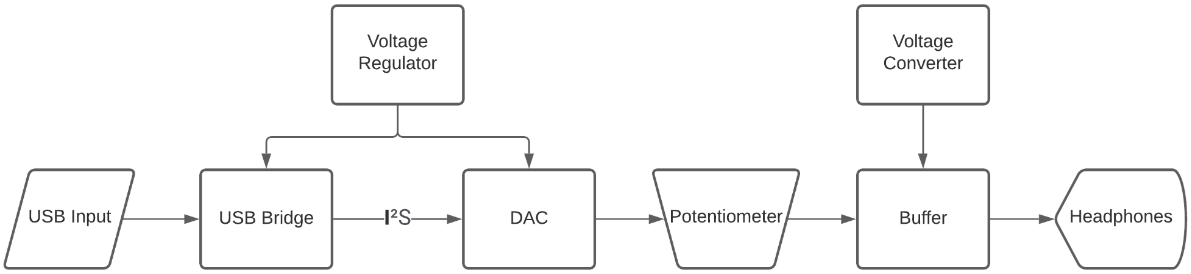


Figure 1: USB Headphone Amplifier Block Diagram

I decided to base the headphone amplifier on the Texas Instruments PCM2706 USB bridge IC and the PCM5102A DAC. Although the PCM2706 is an all-in-one chip with a built-in DAC and headphone output, I wanted to minimize compromises to sound quality while also making this project a challenge, which is why I chose to use an external DAC. I checked the datasheets of both ICs to confirm that they operate at 3.3V and use the same I^2S data format. In this case, they both use the left-justified format. This reassures that both of the ICs are compatible with each other and can be used in the project. I used the NCP612 LDO regulator from OnSemi to generate 3.3V from the 5V supply for the DAC and the USB bridge. An LDO regulator was used here due to the low dropout voltage condition from going 5V to 3.3V. An added bonus of the LDO is that there is no switching noise with the cost of power dissipation. The use of the LME49600 high current headphone buffer enables the capability to drive over the ear headphones. The load current is calculated to be $14.7\text{mW}/2.1\text{V}_{\text{RMS}}$ = 7mA. Using this information and the datasheet for the LME49600TS, the upper swing of the voltage output is limited to approximately +5V - 1.5V = 3.5V, and the lower swing is limited to -5 + 1.5V = -3.5V. This leaves an extra margin since the signal coming from the DAC is approximated to be ± 3V_{P}.

The -5V that is supplied to the LME49600 buffer comes from the LTC1046 switched-capacitor voltage converter IC. The PCM5102A DAC outputs a ground-centered 2.1V_{RMS} waveform and as a result, doesn't require the traditional DC blocking capacitors and a muting circuit. This is ideal because it allows for greater swing and the DC blocking capacitors would form an undesirable high pass filter that would affect the low-frequency response. Since 2.1 V_{RMS} is slightly smaller than 6 V_{PP}. This is superior to the 0.55 V_{PP} output swing from the internal DAC of the PCM2706.

For the I/O components, I chose a USB type B receptacle for the input and a standard quarter-inch headphone jack for the output. The potentiometer is a 10kΩ logarithmic scale that will control the volume of the output and is placed before the buffer to use the low output impedance from the buffer. The maximum power consumption of the 300Ω headphones is calculated to be around 2.1V_{RMS}/300Ω = 14.7mW. Since the HD600 headphones have a sensitivity rating of 100.57 dB/mW in power, we can drive these headphones without any issue. Given this rating, a little less than a mW is needed for listening.

Needed to ensure that the circuit does not exceed the rating for a USB 2.0 port (5V 500mA). I calculated the approximate quiescent current by looking at the datasheets of the components to see what the complete circuit would draw at idle.

3. PCB Design in Altium

The process involved creating a schematic component library that includes the schematic symbols for all the components used in the design. I followed the recommended configurations for each of the ICs from their respective datasheets such as the decoupling capacitors and also added in electrolytic capacitors and resistors to suppress noise. I included jumper resistors to enable/disable different modes of operation for the DAC. Specifically, I made it so that I can select whether the DAC utilizes a low latency IIR or a normal latency FIR digital interpolation filter. I also included jumper resistors to be able to select whether the DAC generates its own system clock (SCK) using its internal PLL or it uses the SCK generated from the USB bridge IC. This was made to ease the process of testing whether which configuration resulted in a more preferred sound signature.

I designed the PCB using Altium Designer, which includes making PCB footprints and schematic symbols for various components in the design and performing manual routing.
I used a standard 2-layer PCB and used the entire bottom layer as a ground plane. The next challenge after completing the schematic drawing was to think about where to place the main components for the PCB. I grouped the analog components separately from the digital components to minimize interference. I used polygon pours and vias to create low impedance return paths for the ICs such as the USB bridge and the DAC. I placed decoupling capacitors as close to their respective ICs as possible and added a ground trace in between the left and right channel of the headphone output trace for shielding. I also made labeled test points in the PCB which would make taking measurements more convenient. I generated the Gerber files after completing the PCB design phase. I chose OSHPARK for the fabrication house and made sure to follow their drill specification rules.

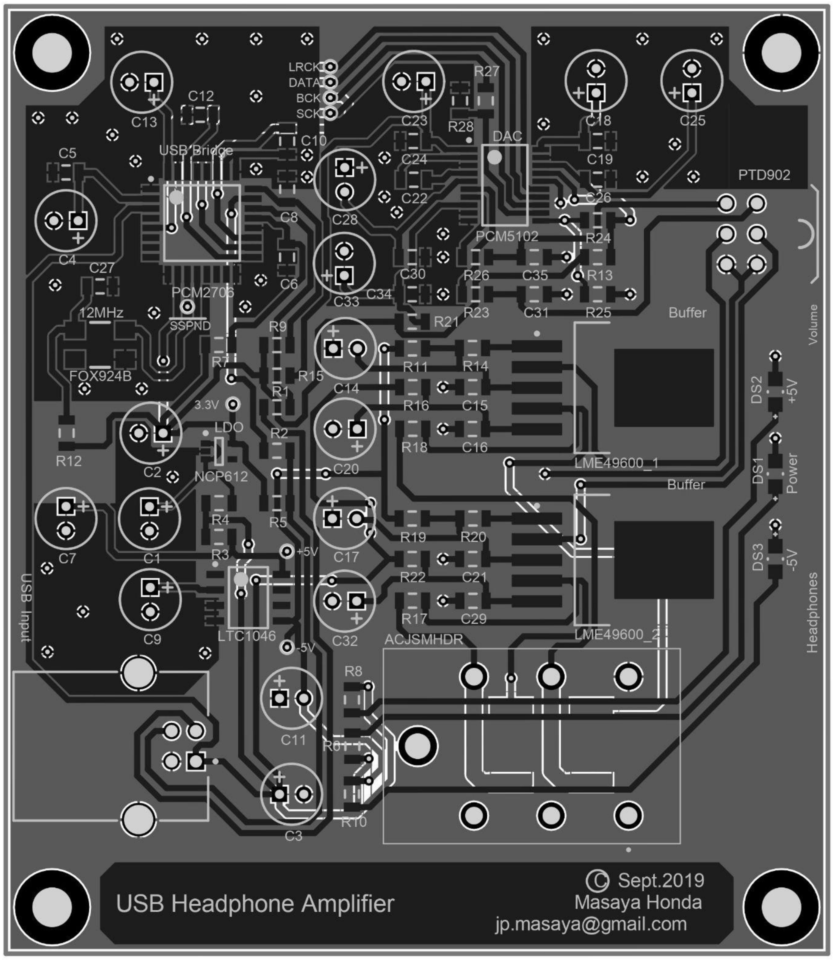


Figure 2: PCB

4. Assembly

Upon receiving my PCB in the mail from the fabrication house, I checked to make sure there are no major issues or mistakes in the PCB before soldering the components by performing continuity checks. I soldered the smaller more difficult components first and saved the bigger through-hole components for last.

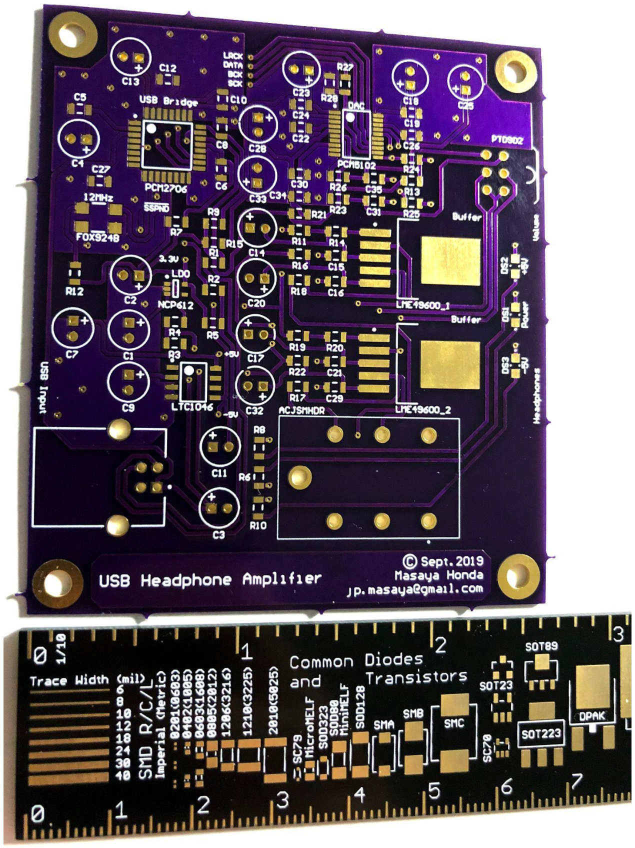


Figure 3: PCB

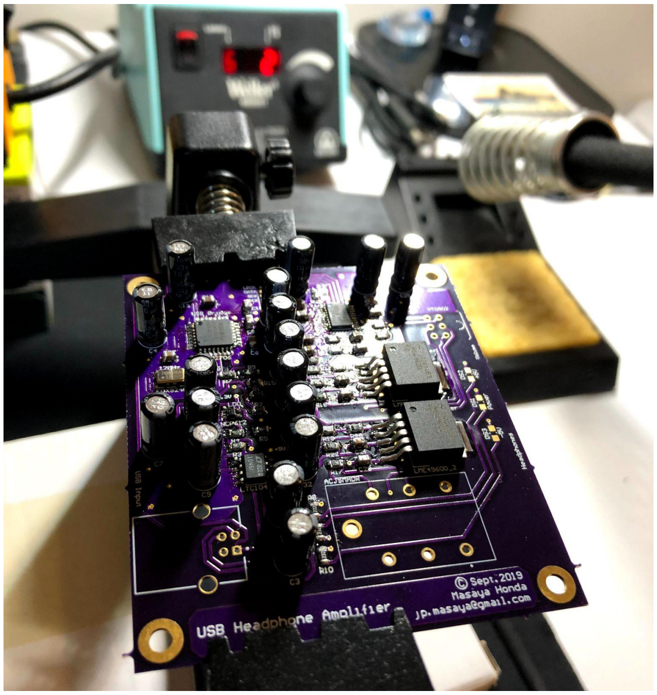


Figure 4: Soldering Components

5. Troubleshooting

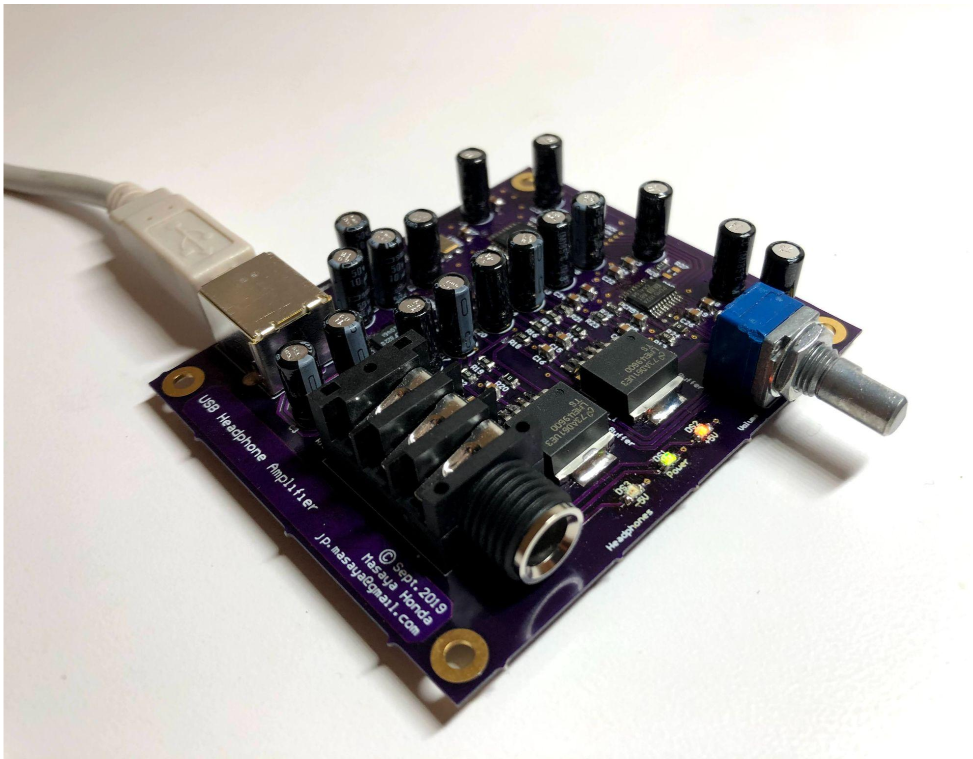


Figure 5: Nonfunctional State

After soldering all of the components, I checked to make sure none of the component's pins were shorted and plugged in the USB from my laptop. The operating system did not recognize the USB device and consequently, the headphone amplifier was not working correctly. I measured the voltages at various pins to make sure that the +5V, -5V, and 3.3V were being generated correctly. The -5V pin was not generated correctly and this was due to an overlooked mistake in the schematic. The CAP- pin on the LTC1046 was accidentally grounded along with the ground pin. As shown above, the -5V LED is not on, which confirms the mistake. To be able to use an external power supply to create -5V, I needed to desolder the LTC1046 chip that is responsible for making -5V. Even after solving the -5V issue, the computer still did not recognize the device. Taking a look at the schematic and the datasheets for each IC revealed another error related to the USB bridge IC. There are two modes that this chip needs to be configured in for power, bus-powered or self-powered. The issue was that not all of the pins were configured for bus-powered operation, which is a requirement to power the device with USB power. To fix the issue of not configuring certain pins to be in bus-powered mode, I made a modification to the circuit by using a combination of wires and desoldering pins.

The PCB after all the issues were fixed is shown below. The red alligator clip is responsible for supplying -5V, which can be confirmed with the -5V LED indicator that is on. With these issues fixed, the headphone amplifier worked as intended when plugged into the computer. Afterward, I soldered the LTC1046 5V to -5V converter IC back on to eliminate the use of an external -5V power supply. These fixes were updated in the schematic and PCB files in Altium. The revised PCB in Altium is shown below as well.

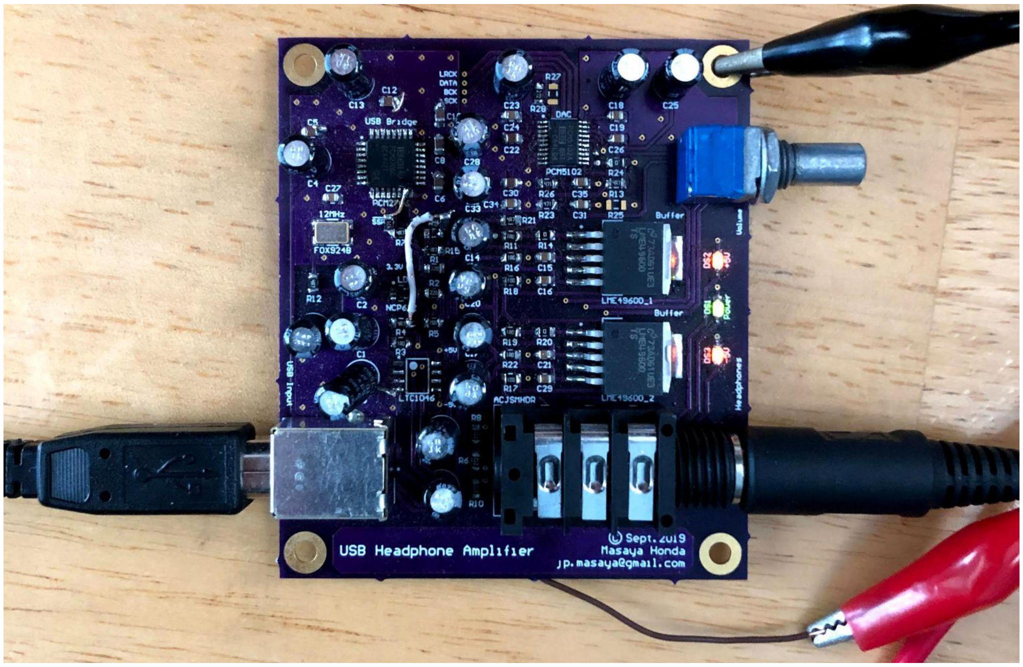


Figure 6: Working with an external -5V supply

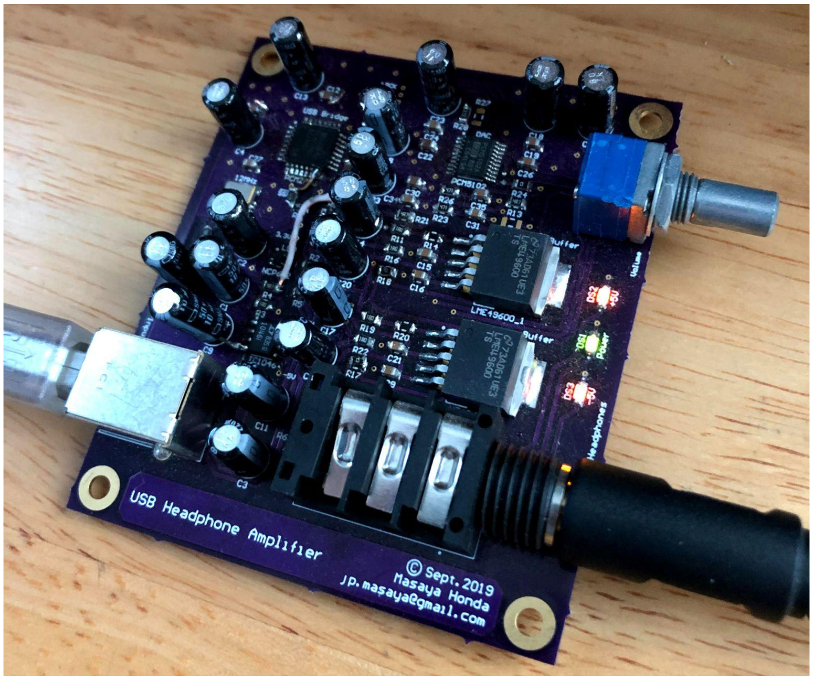


Figure 7: Working with an internal -5V supply

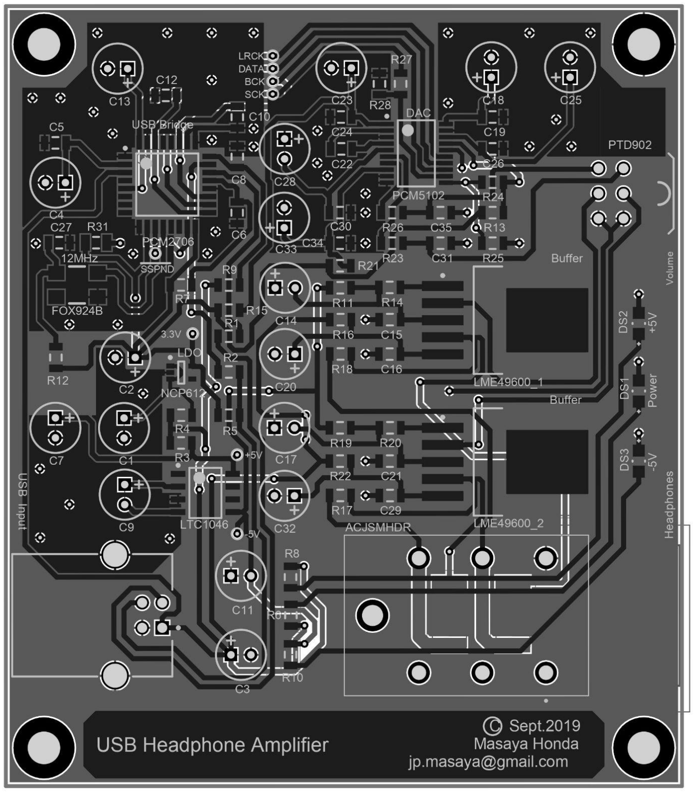
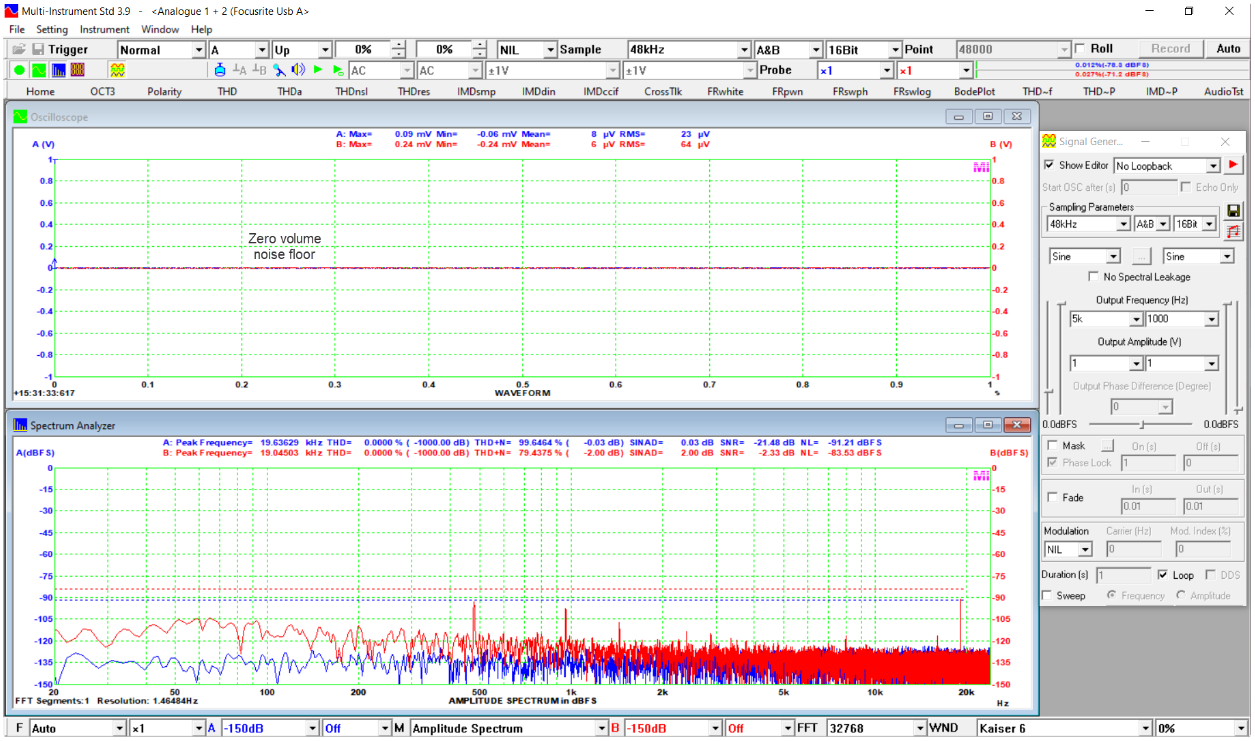


Figure 8: Revised PCB

6. Test Run and Analysis

Although I was unable to hear any undesired background noise when using this headphone amplifier, I decided to measure the noise floor using a spectrum analyzer. The plot in blue represents the system level noise and the headphone amplifier is represented in red. I measured the noise floor of the headphone amplifier under the condition of no input signal and with a 1kHz sine input as shown below.



Multi-Instrument Std 3.9 - <Analogue 1 + 2 (Focusrite Usb A>

Trigger: Normal, A, Up, 0%, 0%, NIL; Sample: 48kHz, A&B, 16Bit; Point: 48000; □ Roll

AC, AC, ±1V, ±1V; Probe: ×1, ×1

0.012%(-78.3 dBFS)
0.027%(-71.2 dBFS)

Oscilloscope

A: Max= 0.09 mV Min= -0.06 mV Mean= 8 µV RMS= 23 µV
B: Max= 0.24 mV Min= -0.24 mV Mean= 6 µV RMS= 64 µV

+15:31:33:617

Zero volume noise floor

Spectrum Analyzer

A: Peak Frequency= 19.63629 kHz THD= 0.0000 % (-1000.00 dB) THD+N= 99.6464 % (-0.03 dB) SINAD= 0.03 dB SNR= -21.48 dB NL= -91.21 dBFS
B: Peak Frequency= 19.04503 kHz THD= 0.0000 % (-1000.00 dB) THD+N= 79.4375 % (-2.00 dB) SINAD= 2.00 dB SNR= -2.33 dB NL= -83.53 dBFS

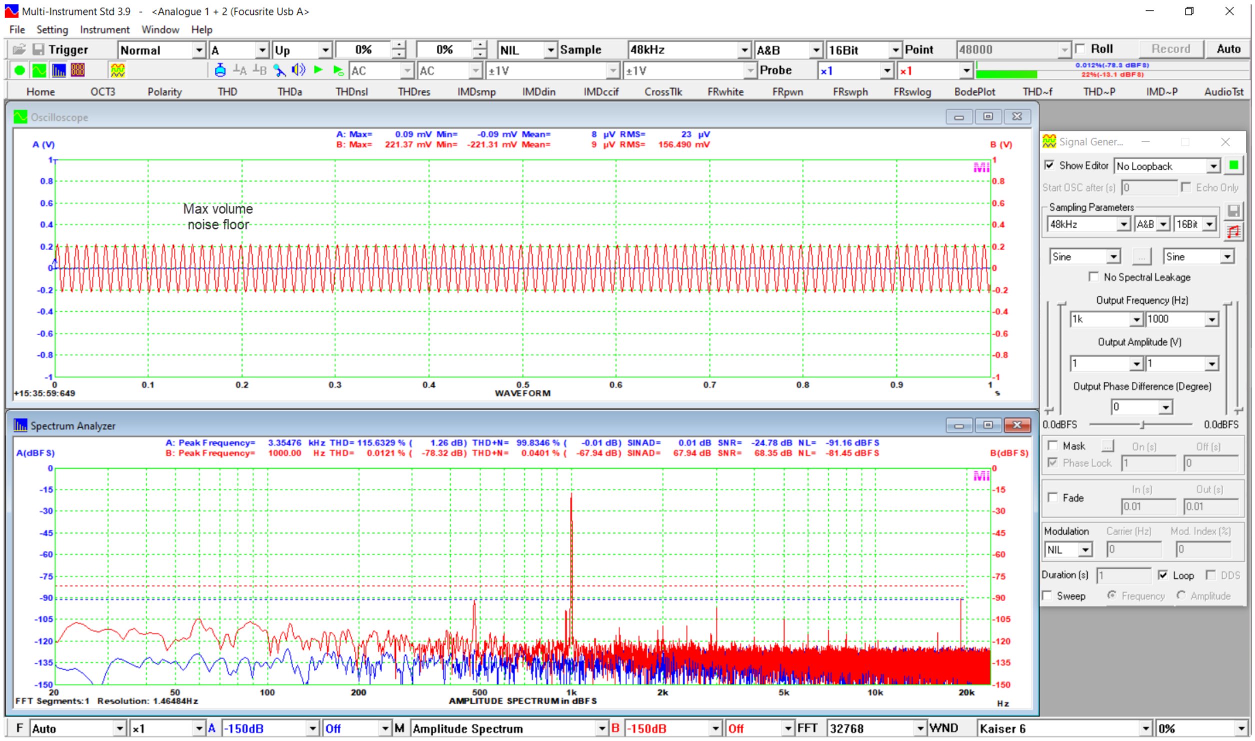
FFT Segments:1 Resolution: 1.46484Hz

F Auto, ×1; A -150dB, Off; M Amplitude Spectrum; B -150dB, Off; FFT 32768; WND Kaiser 6; 0%

Signal Gener...

☑ Show Editor: No Loopback
Start OSC after (s): 0 □ Echo Only
Sampling Parameters: 48kHz, A&B, 16Bit
Sine, Sine
□ No Spectral Leakage
Output Frequency (Hz): 5k, 1000
Output Amplitude (V): 1, 1
Output Phase Difference (Degree): 0
0.0dBFS 0.0dBFS
□ Mask On (s): 1 Off (s): 0
☑ Phase Lock
□ Fade In (s): 0.01 Out (s): 0.01
Modulation: NIL Carrier (Hz): 0 Mod. Index (%): 0
Duration (s): 1 ☑ Loop □ DDS
□ Sweep ● Frequency ○ Amplitude

Figure 9: Noise floor with no input signal

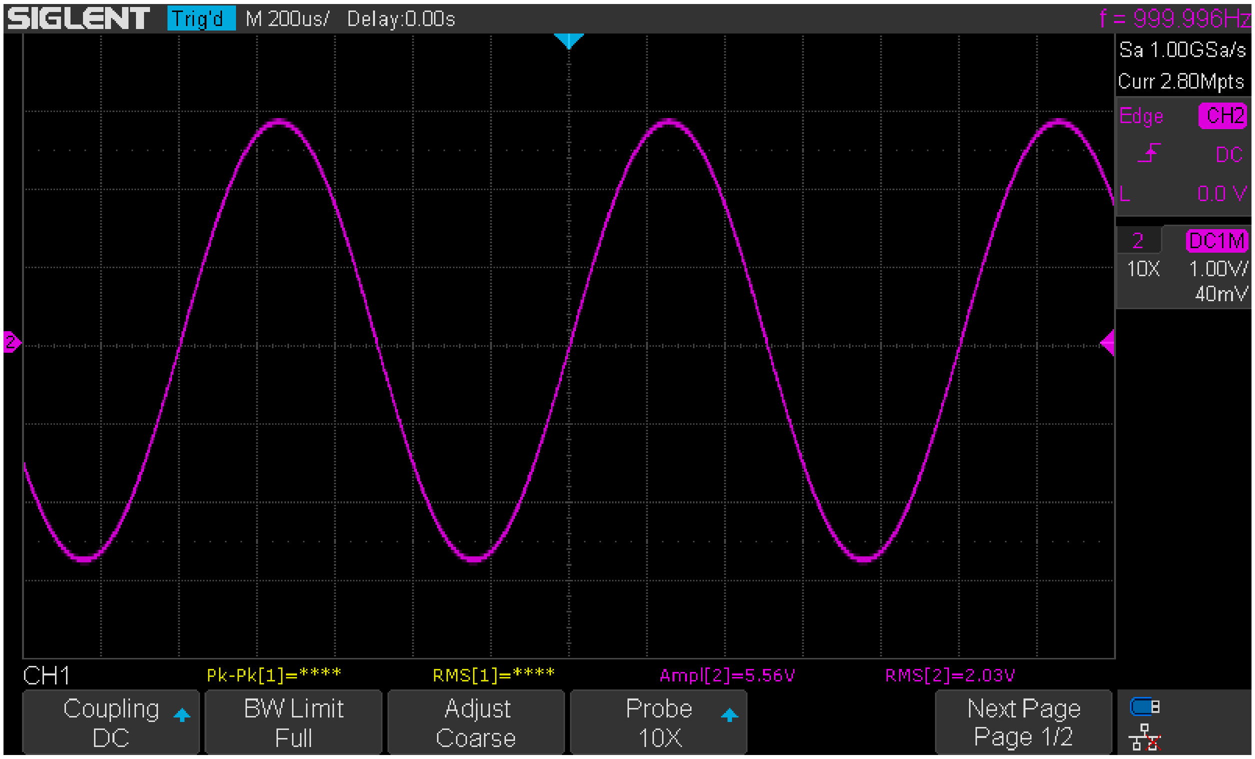

Multi-Instrument Std 3.9 - <Analogue 1 + 2 (Focusrite Usb A>
File Setting Instrument Window Help
Trigger Normal A Up 0% 0% NIL Sample 48kHz A&B 16Bit Point 48000 Roll Record Auto
AC AC ±1V ±1V Probe x1 x1
Home OCT3 Polarity THD THDa THDnsl THDres IMDsmp IMDdin IMDccif CrossTlk FRwhite FRpwn FRswph FRswlog BodePlot THD~f THD~P IMD~P AudioTst
Oscilloscope
A (V)
B (V)
Max volume
noise floor
+15:35:59:649
WAVEFORM
Spectrum Analyzer
A(dBFS)
B(dBFS)
FFT Segments:1 Resolution: 1.46484Hz
AMPLITUDE SPECTRUM in dBFS
Hz
Signal Gener...
Show Editor No Loopback
Start OSC after (s) 0 Echo Only
Sampling Parameters 48kHz A&B 16Bit
Sine Sine
No Spectral Leakage
Output Frequency (Hz) 1k 1000
Output Amplitude (V) 1 1
Output Phase Difference (Degree) 0
0.0dBFS 0.0dBFS
Mask On (s) Off (s)
Phase Lock 1 0
Fade In (s) Out (s) 0.01 0.01
Modulation Carrier (Hz) Mod. Index (%) NIL 0 0
Duration (s) 1 Loop DDS
Sweep Frequency Amplitude
F Auto x1 A -150dB Off M Amplitude Spectrum B -150dB Off FFT 32768 WND Kaiser 6 0%

A: Max= 0.09 mV Min= -0.09 mV Mean= 8 µV RMS= 23 µV
B: Max= 221.37 mV Min= -221.31 mV Mean= 9 µV RMS= 156.490 mV

A: Peak Frequency= 3.35476 kHz THD= 115.6329 % (1.26 dB) THD+N= 99.8346 % (-0.01 dB) SINAD= 0.01 dB SNR= -24.78 dB NL= -91.16 dBFS
B: Peak Frequency= 1000.00 Hz THD= 0.0121 % (-78.32 dB) THD+N= 0.0401 % (-67.94 dB) SINAD= 67.94 dB SNR= 68.35 dB NL= -81.45 dBFS

Figure 10: Noise floor with 1kHz sine input

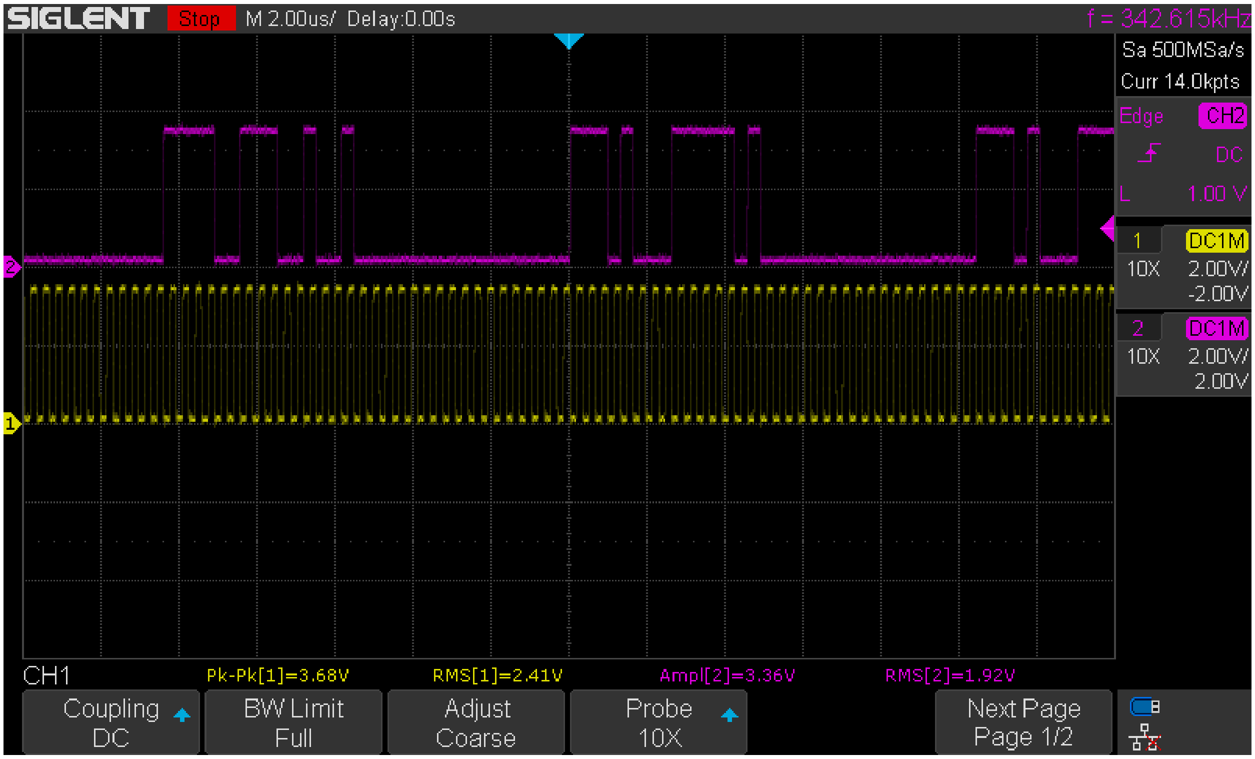
Furthermore, I measured the output of the headphone amplifier with the volume set to its maximum given a 1kHz sine wave at the input. The figure below shows a clean sinusoidal waveform at the output. I also probed the DATA, BCK, and LRCK pins to see their respective waveforms for the I^{2}S being sent from the USB bridge to the DAC.


SIGLENT Trig'd M 200us/ Delay:0.00s f = 999.996Hz

Sa 1.00GSa/s
Curr 2.80Mpts
Edge CH2 DC
L 0.0 V
2 DC1M 10X 1.00V/ 40mV

CH1 Pk-Pk[1]=**** RMS[1]=**** Ampl[2]=5.56V RMS[2]=2.03V

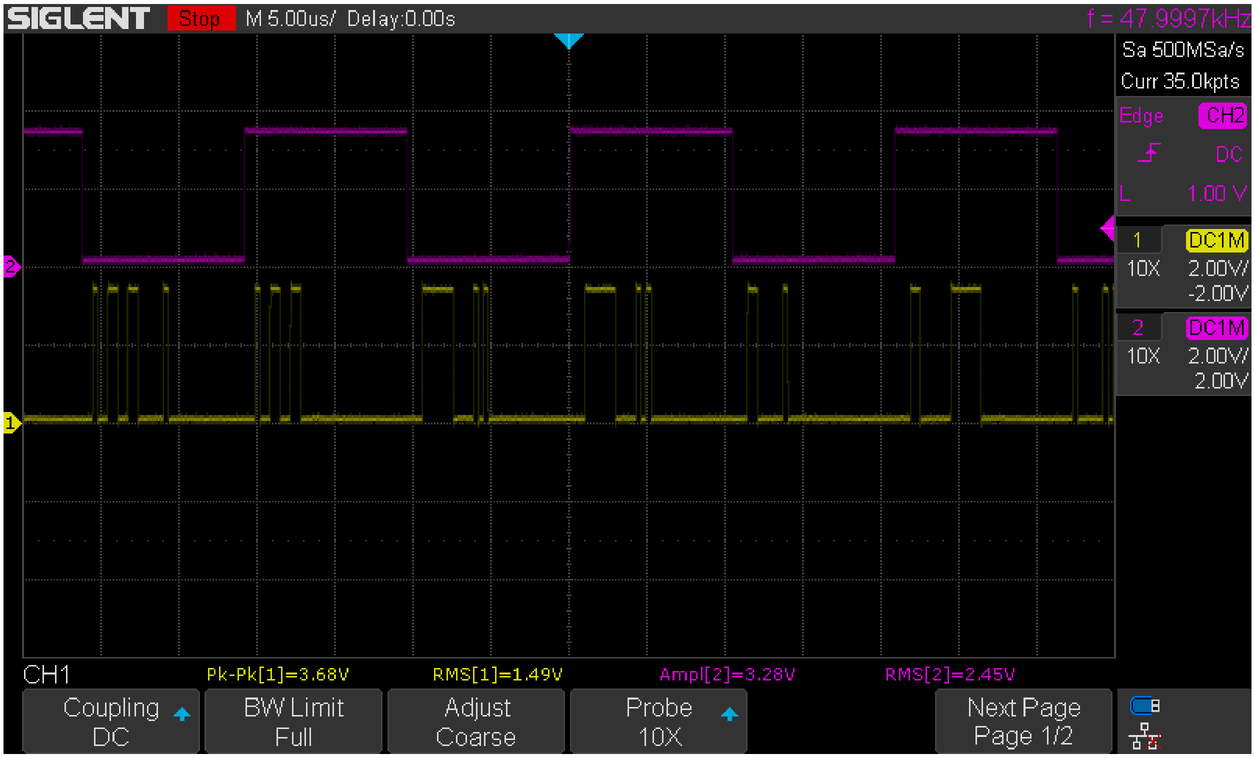
Figure 11: 1kHz sine probed at the headphone output


SIGLENT Stop M 2.00us/ Delay:0.00s f = 342.615kHz

Sa 500MSa/s
Curr 14.0kpts
Edge CH2 DC
L 1.00 V
1 DC1M 10X 2.00V/ -2.00V
2 DC1M 10X 2.00V/ 2.00V

CH1 Pk-Pk[1]=3.68V RMS[1]=2.41V Ampl[2]=3.36V RMS[2]=1.92V

Figure 12: BCK (CH1) and DATA (CH2)



Stop M 5.00us/ Delay:0.00s f = 47.9997kHz

Sa 500MSa/s
Curr 35.0kpts

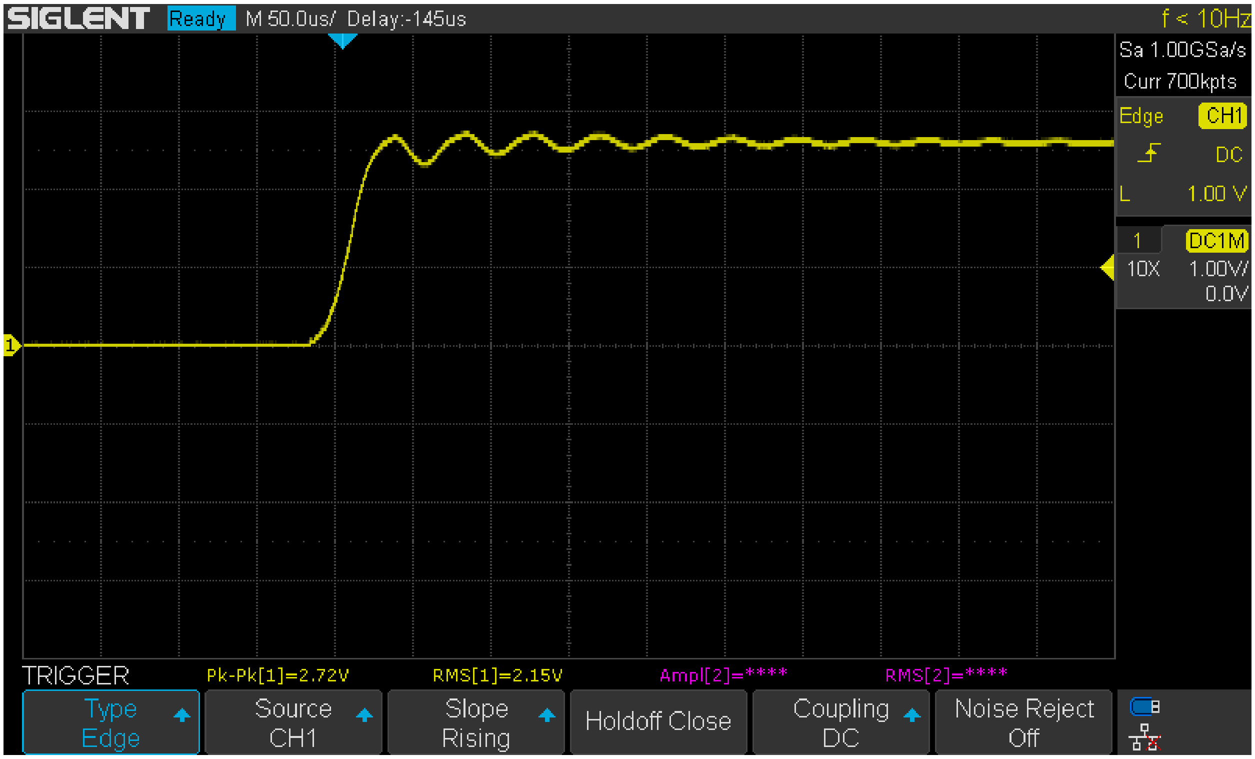
Edge CH2
DC
L 1.00 V

1 DC1M 10X 2.00V/ -2.00V

2 DC1M 10X 2.00V/ 2.00V

CH1 Pk-Pk[1]=3.68V RMS[1]=1.49V Ampl[2]=3.28V RMS[2]=2.45V

Figure 13: DATA (CH1) and LRCK (CH2)



Ready M 50.0us/ Delay:-145us f < 10Hz

Sa 1.00GSa/s
Curr 700kpts

Edge CH1
DC
L 1.00 V

1 DC1M 10X 1.00V/ 0.0V

TRIGGER Pk-Pk[1]=2.72V RMS[1]=2.15V Ampl[2]=**** RMS[2]=****

Figure 14: Step Response

7. Conclusion

Most of the material I learned throughout my classes is related to the transistor level of ICs, so it was interesting to see how each of these ICs functions together to create a working usable device for the end-user. Also, I learned techniques to minimize noise and interference, as this was the first project where I designed a PCB to this level of complexity. During the PCB design process, I gained a new perspective that I was not aware of when coming up with the design of the project. Specifically, I was not aware of how important component placement and manual trace routing is when trying to minimize noise and impedances in vital parts of the design. Thus, I found the significance of being aware of the hidden nonidealities that were not apparent during the initial development of the project when approaching the design from a systems point of view. I realized that viewing the project from a broad systems level requires one to not only understand the theoretical aspect, but also the underlying challenges that come with implementing real-world applications. As a result, this project has been immensely insightful in giving me a new perspective of viewing devices as a system consisting of interworking functionalities, instead of only looking at the individual functionalities of the device.