CS140E Final Project

March 21, 2024

Isabel Berny Griffin Miller

Summary

We set up a MAX98357A amplifier connected to a speaker in addition to a SPH0645LM4H-B microphone using the I2S protocol. Getting the amplifier to work over I2S has been an open problem in CS140E, and we are now able to successfully play an mp3 file of arbitrary size through the raspberry pi.

I2S Microphone

Setting up the I2S microphone was fairly straightforward by following the CS240LX lab. We were successfully able to record audio by taking samples at 44.1KHz and using a Fourier transform to detect the dominant frequencies. One thing we did initially struggle with was the microphone only being able to detect audio extremely close to it (up to ~1 cm away). We eventually realized that by multiplying each sample by a constant before applying the Fourier transform, we could extend the range that the microphone could pick up sound from as samples would not accidentally be zeroed out throughout the process. Multiplying by a constant does not affect the frequency read as we rely on motion rather than exact values.

Bitbang Speaker

The I2S lab gave us a solid foundation to start investigating the speaker. However, we were severely roadblocked when we were unable to get the TX FIFO to empty. As an intermediate step (partially to test the speaker itself as we had to solder the amplifier ourselves and did not feel confident in our soldering skills), we were successfully able to output a 440 Hz square wave through bitbanging.

Specifically, we calculated the half-period of this wave, and set the GPIO DIN pin high and low for ~1000 ms with delays in between. We extended this to then bitbang a jingle with different delays for different notes.

We then realized that in order to play meaningful samples, we would have to bitbang the PCM frame and bit clocks, which would need delays on the order of nanoseconds, which would not be possible to do.

I2S Speaker

With deep investigation into the BCM2835 ARM Peripherals and the MAX98357A Datasheet, we were able to find resolve our previous issues with the following:

1) The BCLK must be sampled on the rising edge
   → Therefore, you must set BIT 22 in the mode register
2) The FS must start high and then go low to start the correct left/right channel synchronization
   → Therefore, you must set BIT 20 in the mode register

With these fixes, we were able to successfully output samples to our FIFO with both 16 and 32 bit samples with 1 channel. Note that you must change the channel width (and extension bit) to indicate the size of your channel.

We determined that leaving the GAIN pin floating allowed for an appropriate 9dB output, and leaving the SD_MODE pin floating was suitable for 16 and 32 bit samples. We leave the exploration of plugging the GAIN into GROUND/VIN with different resistors as well as dual channel speakers to future work.

Generating Samples

In order to create high-quality samples and read them from the pi, we created a python script to take in an mp3 file, generate a wav file, and then write these samples to a binary file. As 1 second of audio generates 44,100 samples, it would be impossible to directly send meaningful audio to the pi through USB. Therefore, we uploaded the binary file to our SD card and modified the FAT32 read() function to iterate through each cluster and write the samples into the TX FIFO. Therefore, the maximum file size that our speaker can read is determined by the SD card.

Conclusion

In the end, we configured the I2S protocol, set up the microphone, and found the keys to making the amplifier/ speaker work, including creating new FAT32 methods for

reading one cluster at a time. Overall, we are very satisfied with this final project as we found a solution to the open problem of sending data to an amplifier over I2S. We were happy that we could, in real time, report our findings to other members of the class that also wanted to use the I2S speaker for their final project. We additionally extended the 240LX lab to include a speaker portion.