SANTA CLARA UNIVERSITY	Mechatronics 2024	Andy Wolfe

Lab #7 – Signal Generation

I. Objectives

- Learn how to program an SPI device and use the SPI library
- Learn to generate signals with a DAC
- Learn to use an Audio Amplifier

II. Pre-Lab

Preparation:

- Review the MCP4811 datasheet
- Review the PAM8302 datasheet
- Review the class example code in Sine_Example_table8 from the class examples folder and understand it.
- Review the Arduino SPI library documentation and functions. https://www.arduino.cc/reference/en/language/functions/communication/spi/

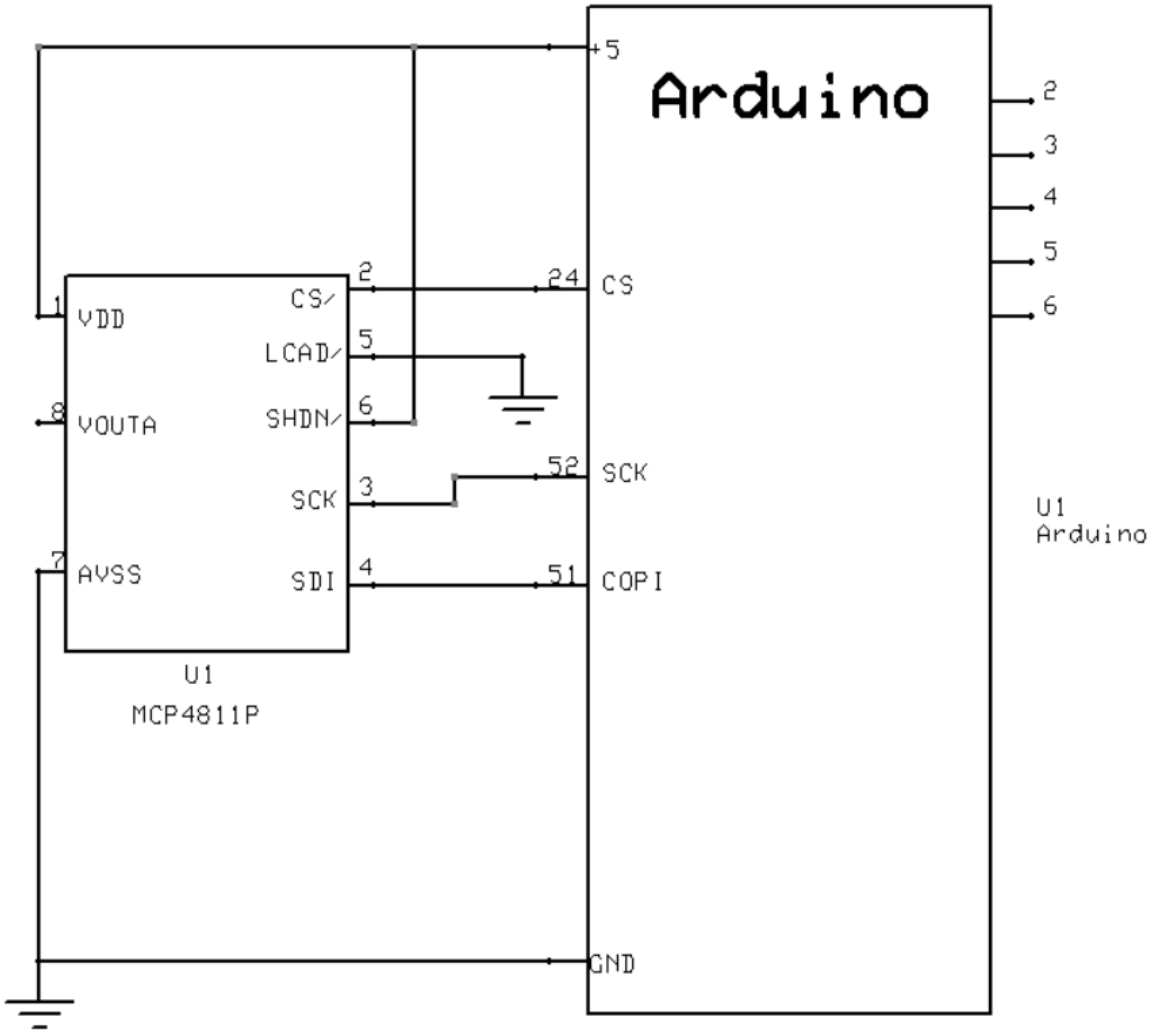
Pre-Lab Report:

- Determine the current sine-wave output frequency of the Sine_Example_table8 from the class examples folder
- Explain why resistors R1 and R3 are used.
- Include a selfie from your planning meeting along with a picture of somebody catching a wave.

III. Lab Procedure 1

Experiment 1:

Connect the MCP4811 D/A converter to your Arduino as shown below.



Connect a voltmeter between MCP4811 pin 8 and ground.

Copy the sketch SPI_DAC_BARE from the lab handouts directory. It should show you how to initialize the SPI interface pins and includes a function to send a DAC value to the chip.

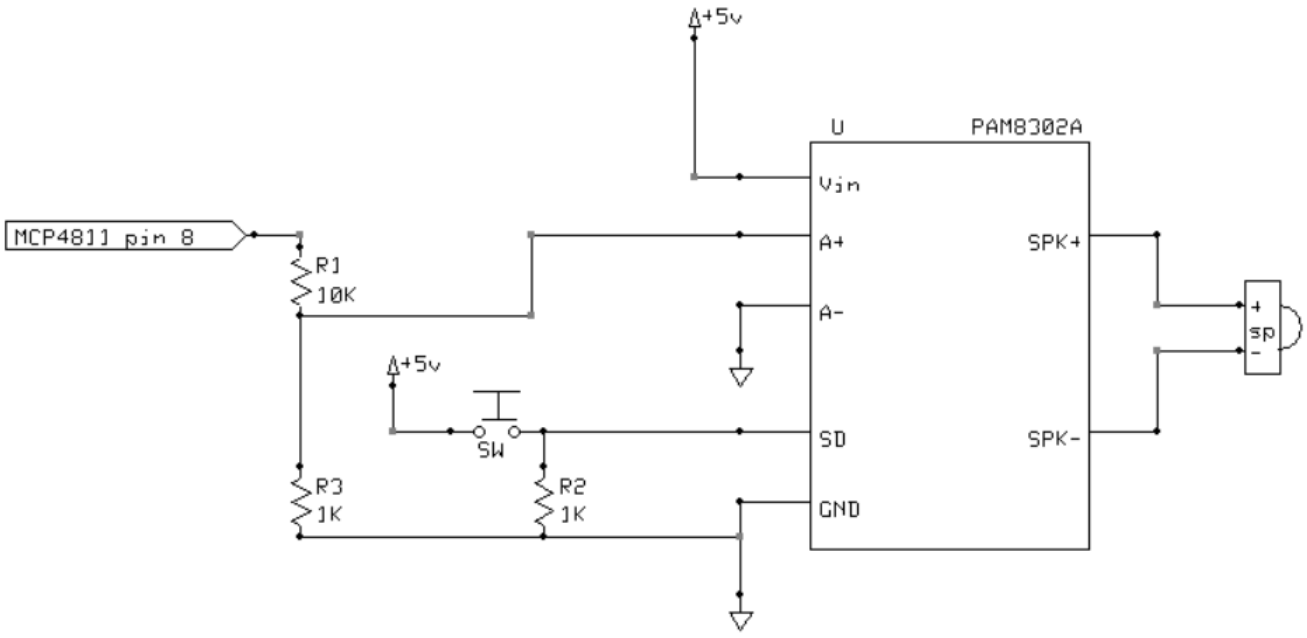
Call the function with an appropriate value to set the DAC output voltage to 2.0 volts. (Demo #1)

Experiment 2:

Next, copy the sketch Sine_Example_table8 from the lab handouts folder. Modify it so that it does not send data out to the serial port but instead sends data to the DAC.
Adjust the program parameters to generate a 40Hz sine wave with 60 steps per cycle. Display this sine wave on the scope. (Demo #2)

Experiment 3:

1) Copy your program and modify the copy so that it outputs a sine wave at 300Hz with 20 steps/cycle. Also, simultaneously output a square wave on Arduino pin 2 at the same frequency. View both on the scope.
2) Connect the audio amplifier module as shown below. The module can fit into your protoboard. Connect the speaker as shown[1]. (Note – you cannot see the sine waves on the amplifier output)
3) When you push the button, you should hear the 300Hz tone.
4) Switch the amplifier input (into R1) from the MCP4811 to pin 2 of the Arduino. You should hear a similar tone. Describe both tones and any differences between them in your report.
5) Borrow a large white speaker from me. Repeat steps 3 and 4. Document any differences in what you hear. (Demo #3 – 300Hz tone on the large speaker)
6) See if you can increase the steps per cycle and still hit 300Hz. See if the tone improves.



[1] If you have 2 speakers, just connect one.

Experiment 4:

1) Copy your program and modify the copy so that it includes a function:

void mytone(int freq, long tonelength) // frequency in Hz, length in µs

This function will play a note at the frequency provided for the time specified.

The following frequencies approximately correspond to the following notes:

A	55	110	220
B	62	123	247
C	65	130	262
D	73	147	294
E	82	165	330
F	87	175	349
G	98	196	392

Assume a quarter note is 400,000 µs.

Test your routine by having it play some notes.

Then modify your program to play Mary had a little lamb over and over. (at least the first two stanzas)
http://www.enchantedlearning.com/music/sheetmusic/maryhadalittlelamb.shtml

If you can't read the music – find someone who can.

It may sound better if there is a slight pause between notes.

Demo #4