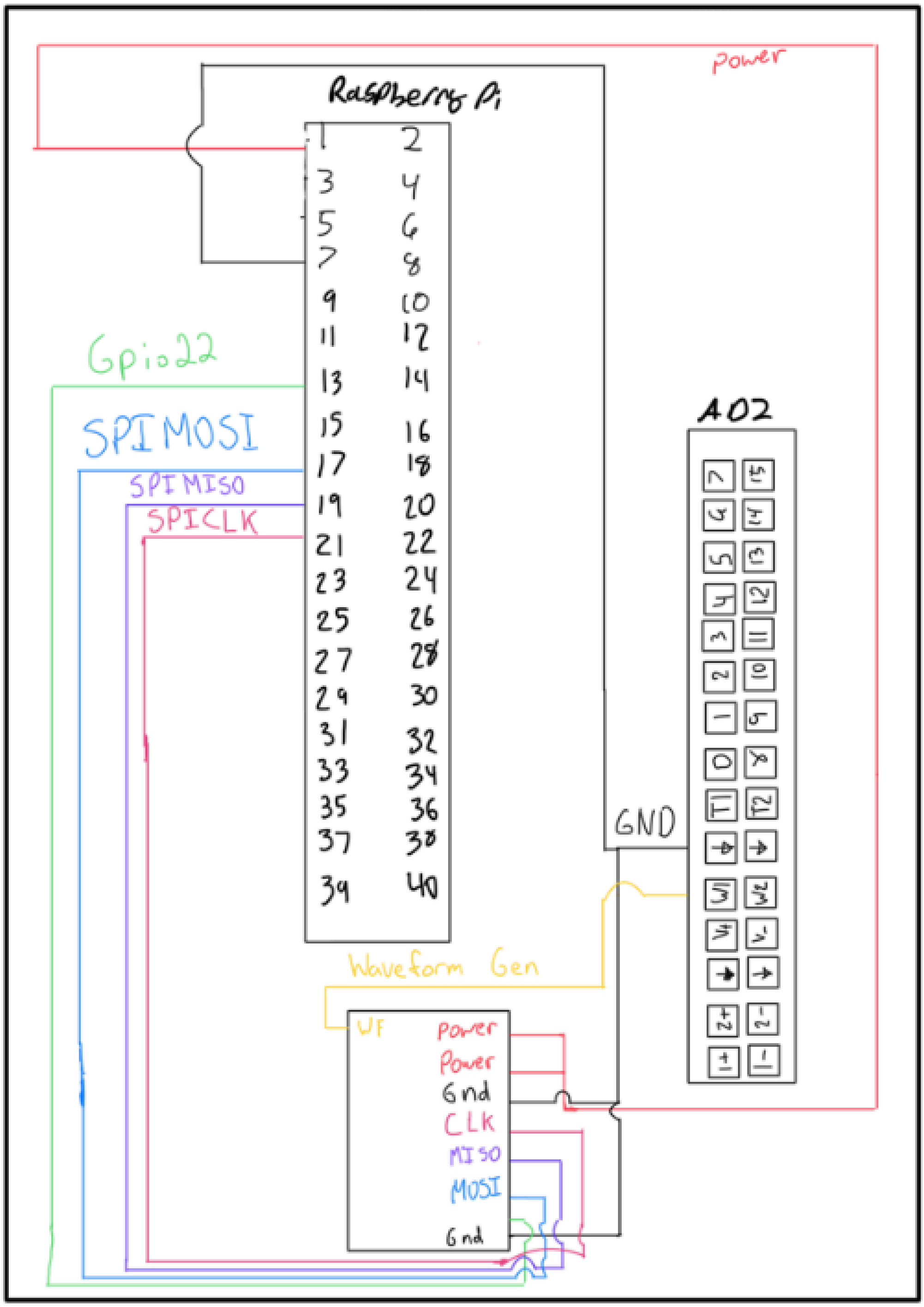
Lab 3: Interfacing with

Samuel Huang
Gregory Miller
Kyle Stallings
CSCE 462 - 501

Hardware Description



The circuit consists of a raspberry pi 400, ADC (MCP3008), and AD2 waveform generator. The Raspberry Pi uses pin 7 as ground for the AD2 waveform generator and ADC. Pins 12,17,19, and 21 and used for GPIO22, SPI MOSI, SPI MISO, and SPI CLK for the ADC respectively. The Raspberry Pi also powers the ADC through pin 1, giving 3.3 Volts to the two upper right pins on the ADC. The

ADC with these connections to the Raspberry Pi takes in an input from the AD2 Wave generator and then communicates a binary version of the input signal to Raspberry pi.

Software Description

The code has a while loop which polls data and calls on two supporting functions to process it. The way that data is polled by this program is that it uses a sample rate, and a number of samples to fill an array called samples. The functions then find the waveform and frequency. The first function is the frequency calculator, which takes in the samples and sample_rate. This function uses the number of zeros, or where the wave is at 0 to determine the frequency. Based on the total amount of zeros, an amount of times zero is found, to determine how many times the wave goes high and low. From here the value is multiplied by a constant of 750 which is the found number that offsets the calculated value from the actual value

The next function in the program is the detect_waveform function, which takes in the calculated frequency and samples. With the polled data, the function checks it against four different conditions. The first condition checks if there is any wave, which is done by seeing if 80 percent of the samples are equal to 0. This 20 percent gap allows for any distortion that is picked up to be ignored and not result in any false waves. The second condition checks to see if it is a square wave, by finding the max and min value and seeing if 85% of values within the samples array fall within ten percent below the max or 10 percent above the min. This accounts for any distortion, while making sure that the values accurately point towards a square wave. The next condition checks to see if it's a sine wave by seeing the amount of consecutive times values are within 20 percent of the maximum value. This number of consecutive times must be greater than the multiple of the period found from the frequency and a constant for the program to determine its a sine wave. The final condition is an else, which determines the wave to be a triangular wave since it's not a square or sine wave.

Overall the software calculates the frequency by the amount of times the waveform hits zero, and determines the waveform based on conditions that rely on the data meeting a certain criteria. With the criteria be adjusted to account for distortion, by using constants or making only a percentage of the data meet the requirements for it to be true. Ultimately the program efficiently and accurately determines the frequency and type waveform.

Lab Questions

1. Summarize the difference between SPI and I2C ports. Explain in what situation using the SPI ports are better than the I2C ports, and vice versa.

SPI and I2C are both used as communication protocols, where SPI is full duplex and I2C is only half-duplex meaning SPI can transfer data simultaneously both ways while I2C can only transmit or receive. SPI uses four data lines, the MOSI, MISI, SCK, and SS which supports higher data rate transfers than the I2C. This means that when a situation requires speed SPI is better suited. I2C uses two data lines, the SDA and SCL, which communicate over the same bus line. This means that if speed doesn't matter too much and multiple devices need to be on the same line, the I2C allows for these connections but with a slower performance.

2. What are the various types of ADCs in use? Which type of ADC is MCP3008 and What are its advantages/disadvantages?

There are the Dual-slope, Sigma-Delta, Successive Approximation, Pipelined, and Flash types of ADCs in use. The MCP3008 is a Successive Approximation type. Its advantages are that it is a middle ground for speed, intuitive to set up with a microcontroller, and is reasonably accurate at a low cost. The disadvantages are that it has a lower resolution and speed when compared to more expensive variants.

3. What is the sampling rate for your oscilloscope?
The highest frequency our oscilloscope could reasonably determine was 1kHz,and from this the sampling rate should be set to two times the max frequency, to accurately reconstruct a signal, making our sampling rate 2000 per second.
4. If you use the same Raspberry Pi to do waveform generation and waveform recognition at the same time, you might generate a waveform that the frequency keeps changing and get random readings from the MCP3008. Explain why this is the case.
The Raspberry Pi does not have enough processing power to accurately generate a waveform and perform the wave and frequency determinations. If both were performed simultaneously the CPU would struggle to run both operations in a timely manner, making the MCP3008 be at random intervals instead of at a consistent period. These large variations in data would create deconstructed signals which would not allow for a reasonable determinations, which is the same reason our max frequency is at 1kHz, is bec the processor can only handle running the program and receiving coherent data up to here.
5. It is highly likely that your sampled data contains lots of noise. How can you filter the noise? Explain your method.
The data contains a decent amount of noise within our working range frequency of 10-200 Hz. A way to filter noise is using a Fast Fourier Transform on a signal to remove the frequencies outside of the generated Frequencies that were expected. This would lead to more accurate results, but with this more processing which may not be allowed with the Raspberry Pi's limited processing power.