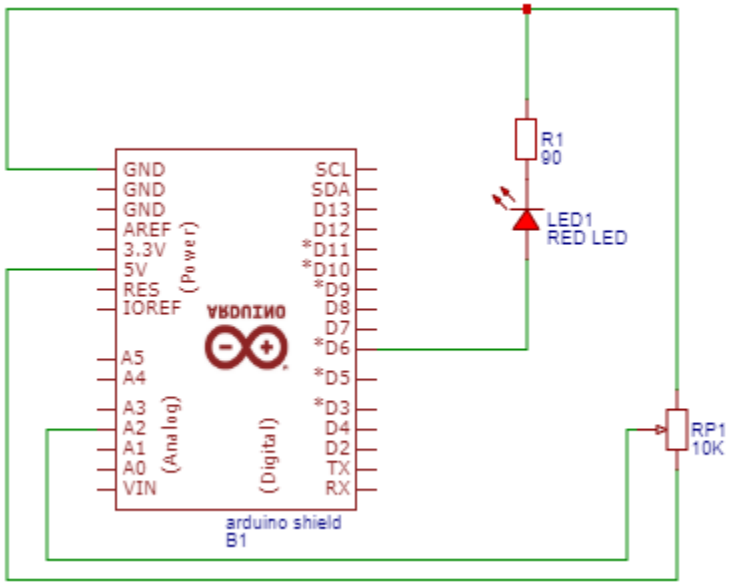
AnalogRead() & AnalogWrite()

Electronic diagram

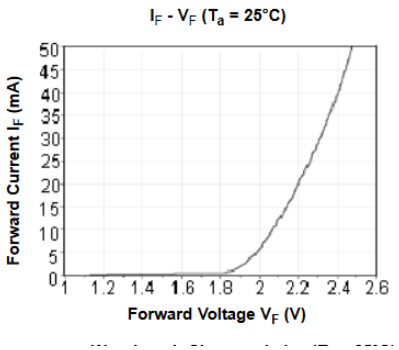


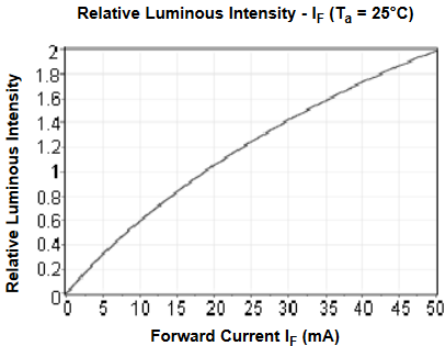
1x Red LED

1x 90 Ω resistor

1x 10k Ω Potentiometer

Luminous Intensity for a red LED





In order to use Ohm's Law, I first looked at the voltage-forward current needed for a luminosity somewhat in the middle of the graph.

Known Variables:

$I_{Supply} = I_{Resistor} = I_{LED} = 30$ mA

$U_{LED} = 2.3V$

$U_{Supply} = 5V$

Equations:

$U_{Resistor} = U_{Supply} - U_{LED} = 5V - 2.3V = 2.7V$

$R_{Resistor} = U_{Resistor} / I_{Resistor} = 2.7 / 0.03 = 90\ \Omega$

Registers used

Port D Data Direction Register

This register was used to set pin 6, which is connected to the LED to output.

Port C Data Direction Register

This register was used to set the analog 2 pin, which is connected to the potentiometer to input.

ADC Control and Status Register B

This register was used to set the analog to digital converter to free-running mode.

ADC Control and Status Register A

This register was used for different tasks, such as, to determine the division factor between the system clock frequency and the input clock to the ADC, enable auto triggering, enable the ADC interrupt, turn on the ADC, and start the ADC conversion.

ADC Multiplexer Selection Register

The ADMUX register was used to set the voltage reference and left adjust the result to allow 8-bit reading.

The Port D Data Register

PORT was used to turn on the LED.

Timer/Counter Control Register A

This register was used to set the output compare pin to compare mode and set the waveform generation mode to fast PWM in order to enable changing the LED brightness.

Output Compare Register A

This register was used to change the brightness of the LED.

Logic Analyzer

Part 1

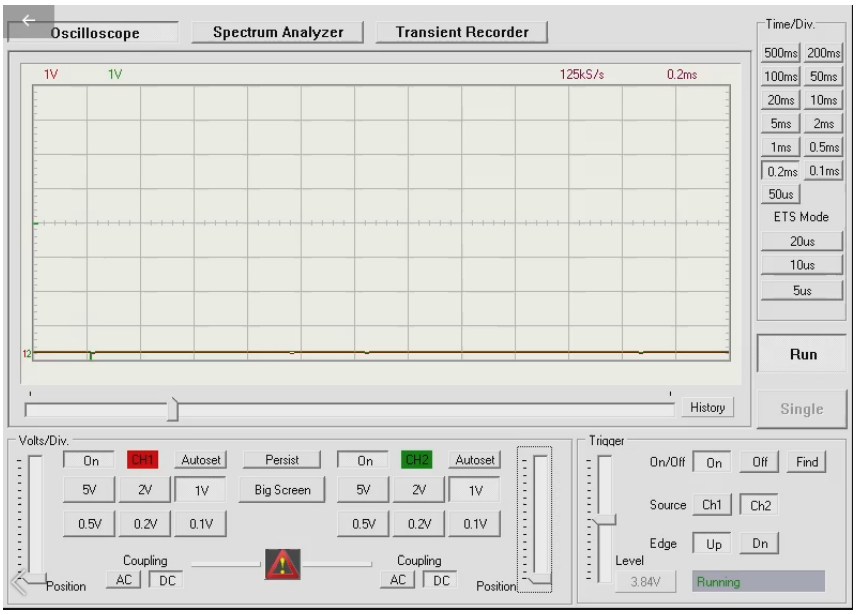


Figure 1 Red: Potentiometer, Green: Arduino output, Potentiometer is turned all the way towards the ground

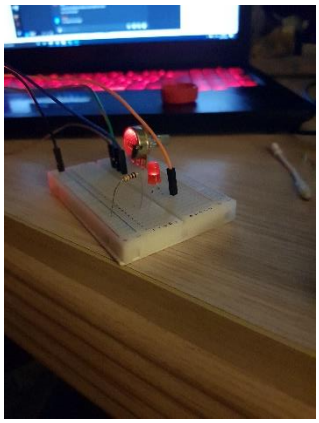


Figure 2 Potentiometer is turned all the way towards the ground

In figure 1 the voltage is almost to 0 and there are small oscillations of the Arduino output. As a result, the output of the LED is very dim.

Part 2

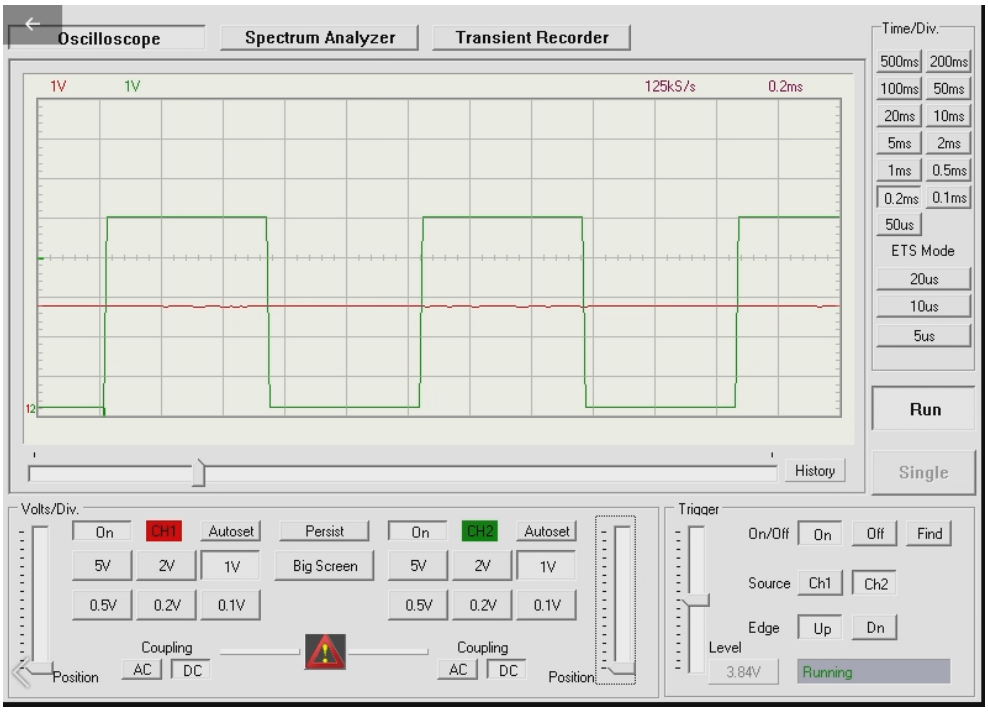


Figure 3 Red: Potentiometer, Green: Arduino output, Potentiometer is turned midway towards the ground and 5V

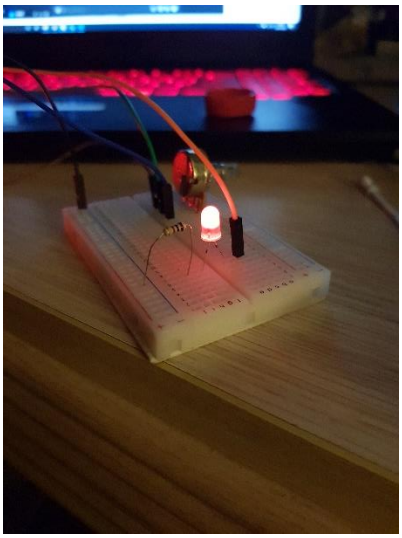


Figure 4 Potentiometer is turned midway towards the ground

As a result, we can see that the frequencies are the same and oscillations of the Arduino output increased making the light of the LED brighter.

Part 3

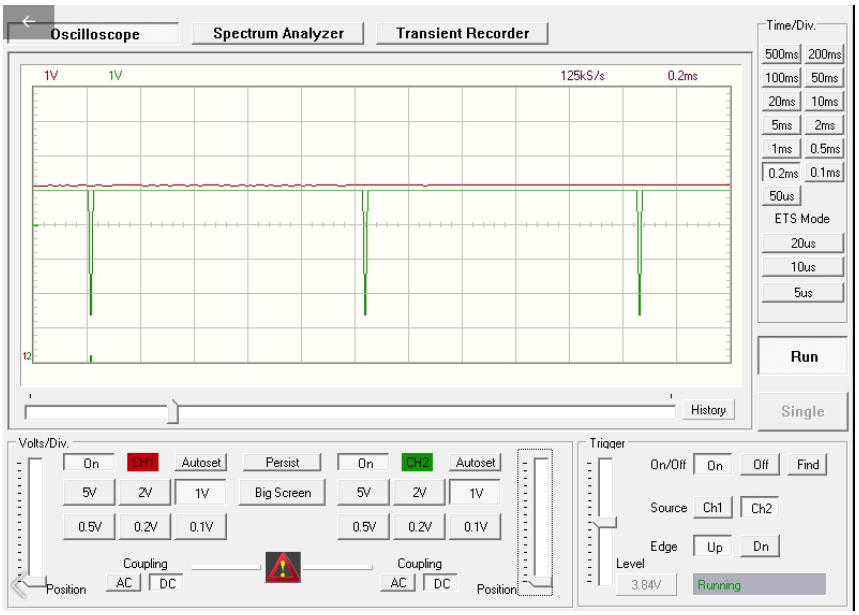


Figure 5 Red: Potentiometer, Green: Arduino output, Potentiometer is turned all the way towards the 5V

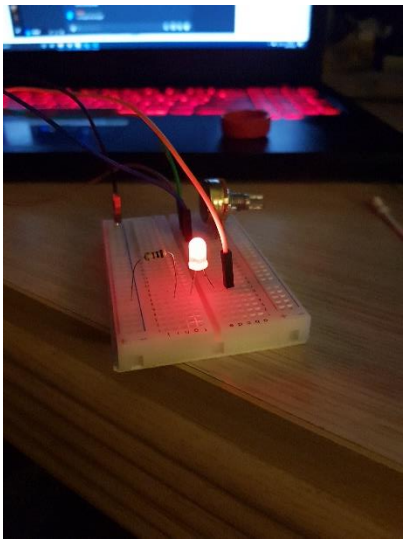


Figure 6 Potentiometer is turned all the way towards the 5V

In figure 5 we notice that the frequency is still the same. The reason why the LED is brighter is the time that it stays on.

Voltage meter

Figure 7 Potentiometer all the way towards the ground

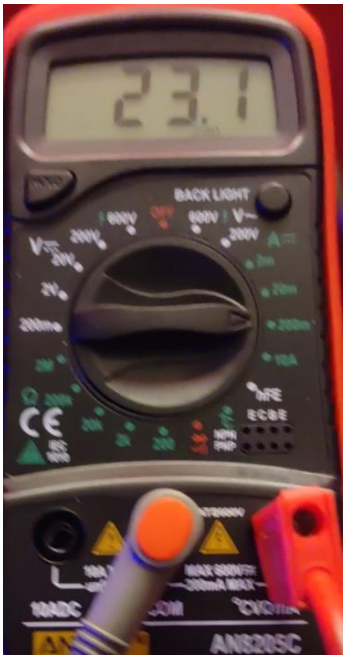


Figure 8 Potentiometer all the way towards the 5V