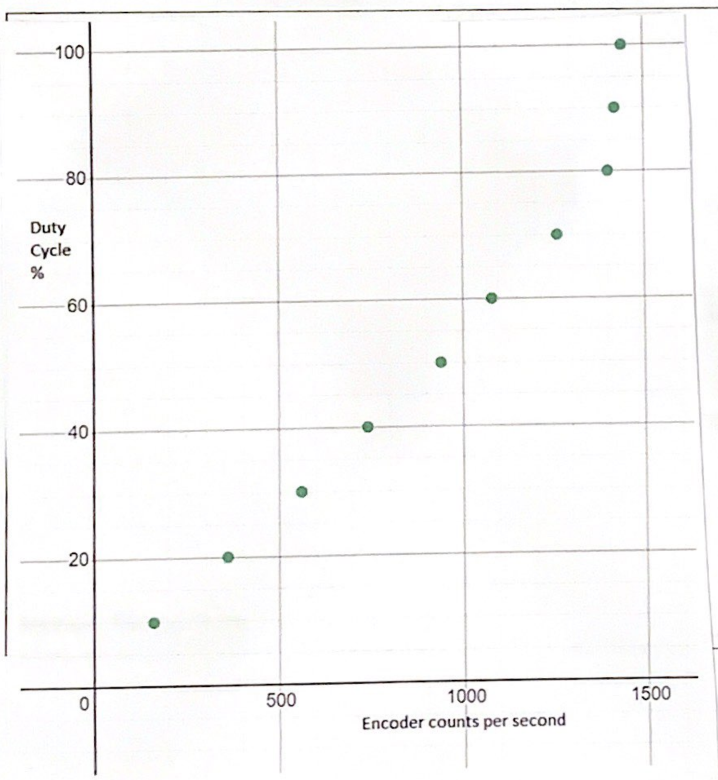
WORK SHEET HERE:

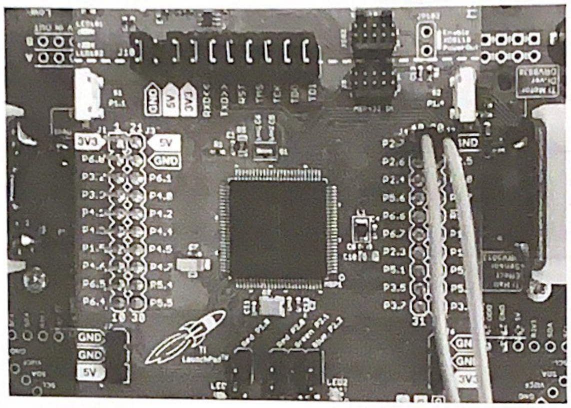
Bin Length: 50 ms (from code)

Duty Cycle %	analogWrite value (0 to 255)	Encoder Counts Per Bin
10	26	8
20	51	18
30	77	28
40	102	37
50	128	47
60	153	54
70	179	63
80	204	70
90	230	71
100	255	72

Plot your data in Excel with Duty Cycle on the y-axis and Encoder Counts **per Second** on the x-axis:

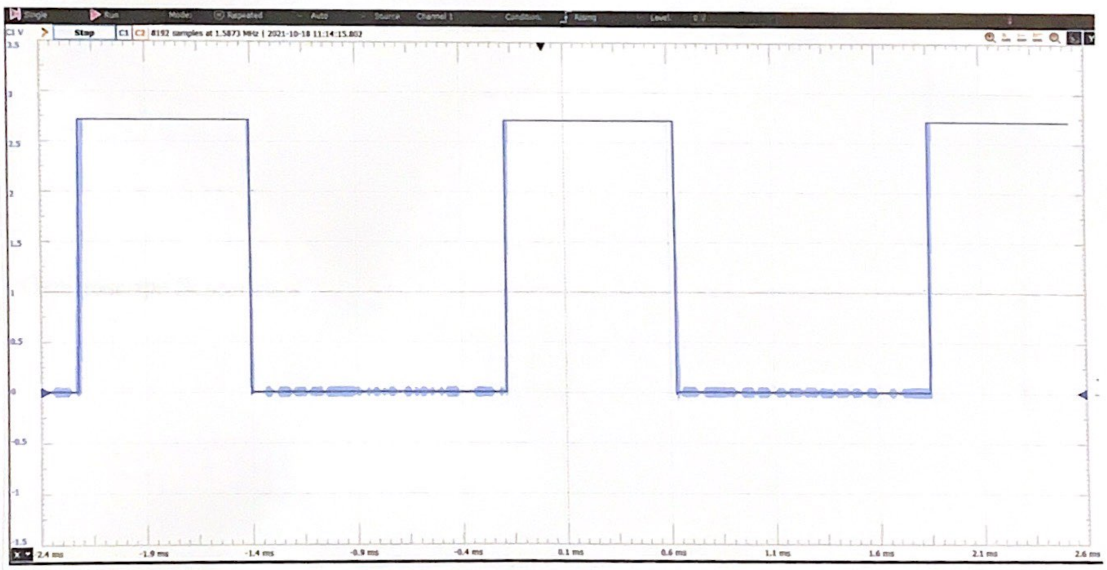


9. Using the skills you've built from previous labs, we can probe the PWM signal the RSLK applies to the left DC motor. Using you AD2, connect Channel 1 to probe the pin PWML. To do this, using the chart above, you must attach 1+ to pin P2.7 and 1− to GND. Notice that you can attach the AD2 directly to the Launchpad (no breadboard necessary). Use the image below for guidance:



AD2 Channel 1 Connected to P2.7 and GND

With the left DC motor running (at a duty cycle of your choice (other than 0% and 100%)), use your AD2's Scope application to view the PWM waveform. Include a screenshot of the waveform here (you must set the time base and amplitude appropriately):



Answer the following questions to verify your oscilloscope view is correct:

Duty Cycle Setting: ___40___ %

Measured Signal On-Time (per period): ___810.4 us___ 810.4us

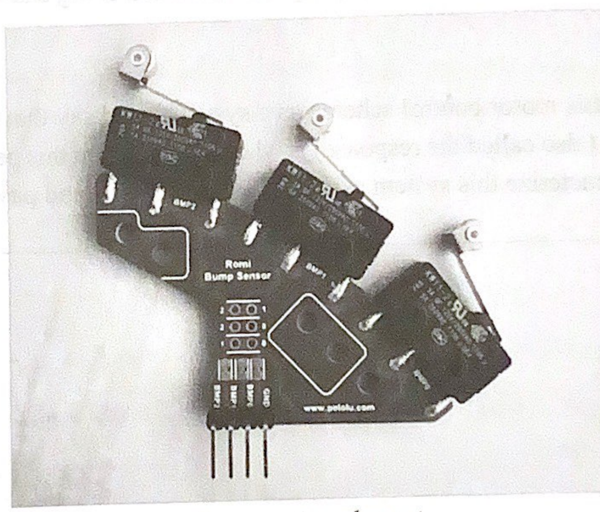
Measured Signal Off-Time (per period): ___1.217ms___ Entire Period (810.4us+1.217ms)

Measured Duty Cycle: ___39.9%___

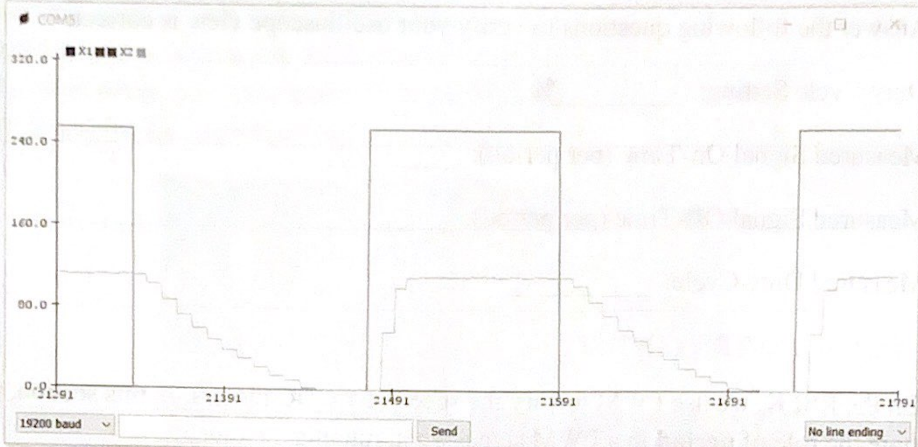
PWM – Period (EXTRA CREDIT)

On the RSLK, we use PWM to vary the speed of the DC motors. In this section, we will demonstrate the role of **period** in a PWM signal when running a motor.

1. Download the sketch pwm_period.ino from CCLE. Launch Energia and open the sketch.
2. Read through the code to get an understanding of what it does. Some of the techniques used here (such as using micros() rather than delay() for time delays) may become useful to you later on.
3. Make sure you have the bumper attachment connected to your car. Look at the following images to ensure this is done correctly:



Bumper Attachment



Serial Plotter Output After Turning the Car On
(Note: your COM port may be different)

8. Press BMP0 to vary the period. You will hear a high frequency warbling sound from the motor for the smaller period. This is because the pulse rate is 10 pulses per second, and the motors can partially follow the pulses at that rate. Compare this pulse rate to then end of the previous exercise, where you measured the PWM pin pulse rate when using analogWrite.

Questions:

1. If we look at this motor control scheme as a system, we'd say that the input is the PWM signal and the output (also called the response) is wheel speed. From this perspective, what kind of filter could you characterize this system as: low pass, high pass, band pass, or band stop? Explain you answer.

It is a low-pass filter. We can deduce this from multiple clues. First, the graph looks like a capacitor – the speed ramps up quickly then slowly dissipates. Secondly, the signal is relatively low. This tells us that it is a low pass filter.

The pwm_period.ino sketch seen here is only for demonstration. As said earlier, when you create your own sketches for the RSLK, you will use the analogWrite(pin, value) function to create a PWM signal.