

# RoboJackets Firmware Training Week 3 Lab Guide

Marine Maisonneuve, Logan Schick

October 5, 2020  
v1.0

## Contents

<b>1</b>	<b>Background</b>	<b>2</b>
1.1	Topics . . . . .	2
1.2	Premise . . . . .	2
1.3	ATMega328P Microcontroller . . . . .	2
1.4	Simulation . . . . .	2
<b>2</b>	<b>Materials</b>	<b>3</b>
<b>3</b>	<b>Objectives</b>	<b>3</b>
3.1	Task 1 - Setup . . . . .	3
3.2	Task 2 - Create Interrupt . . . . .	3
<b>4</b>	<b>Troubleshooting</b>	<b>3</b>
4.1	Solutions . . . . .	3

# 1 Background

## 1.1 Topics

The important topics being discussed this week in lab include datasheets, registers, and Pulse-Width Modulation (PWM).

## 1.2 Premise

The lab premise is to use a push button change the brightness of a LED. This system will use interrupts to change PWM registers to control the power output.

## 1.3 ATmega328P Microcontroller

The Arduino Uno is based on ATmega328P microcontroller which has 3 timer setup controlled using timer registers. We can therefore look at the [datasheet](#) for the microcontroller to figure out the values for those control registers. To save a bit of time skip to section 12-15 (specifically section 15) which explains the timer you'll be controlling in the lab. To complete the lab the only required readings are the register descriptions for the timers, but other sections such as modes of operation will give a better understanding of what these timers can do.

## 1.4 Simulation

If you are using a simulation instead of the hardware, do not worry. The steps are exactly the same. Go to the TinkerCAD link and you will see the circuit that is a subset of the the hardware. The Arduino you see will be what you use, with the LEDs and buttons replicated as they would be on the actual board.

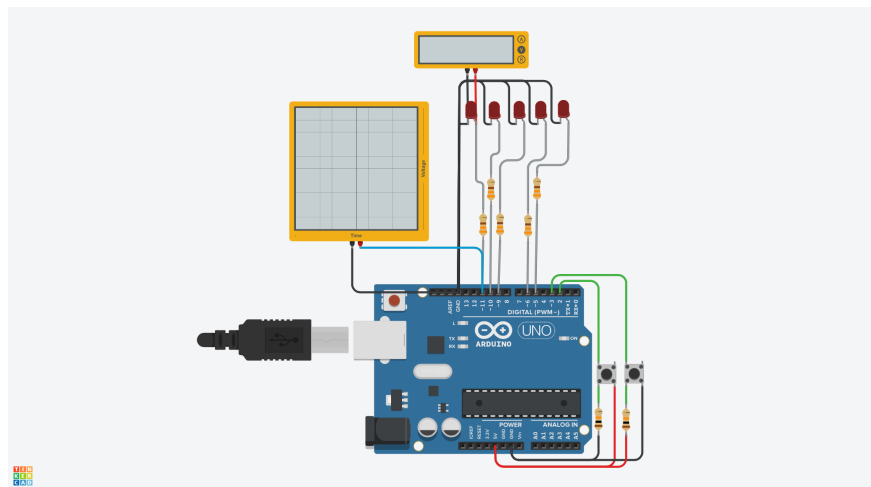


Figure 1: The circuit window of TinkerCAD for this project

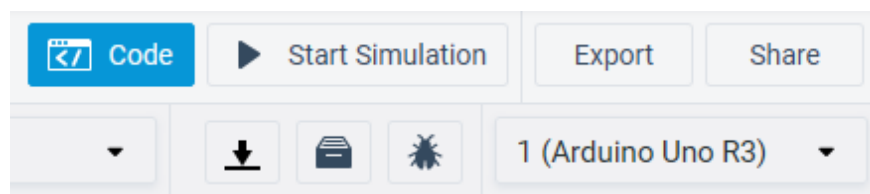


Figure 2: The area which you can use to select your target and compile

## 2 Materials

- [AutoDesk Education Account](#)
- [TinkerCAD](#)

## 3 Objectives

### 3.1 Task 1 - Setup

1. Use the [datasheet](#) to fill in the empty values in the `setup()` function.
  - Output compare can be inverting or non-inverting.
  - The desired timer mode is Fast PWM. The prescaler can be any valid configuration.
  - Note the way the value of the output compare register is set.
2. Fill in TCCR2A value.
  - Set bits based on the composition given at the top of Section 15.11.1 of the datasheet.
  - For information on setting the output compare mode, refer to Section 15.11.1 Table 15-3.
  - For information on setting the timer mode refer to Section 15.11.1 Table 15-8.
3. Fill in TCCR2B value.
  - Set bits based on the composition given at the top of Section 15.11.2 of the datasheet.
  - For information on setting the prescaler, refer to Section 15.11.2 Table 15-9.
  - Once you get the lab up and running try it with the voltage monitor and different prescalers to see how changing the prescaler affects the output.
4. Composition of OCR2A is in Section 15.11.4.

### 3.2 Task 2 - Create Interrupt

1. Have the interrupt method increment the value that goes into the output compare register.
  - Note that when using fast PWM the 8 bit output compare register takes in a value from 0 to 255.
  - Values higher will overflow losing bits above 8, so you may want to account for that.
  - Set up the interrupt to trigger when the button is pushed.

## 4 Troubleshooting

### 4.1 Solutions

We have included the solutions below if you do not complete the lab during the session or if you want to verify your answer. If you need help during the lab ask an instructor!

- [TinkerCAD Solution](#)