



**1ST
MAKER
SPACE**
Design.
Build.
Sustain.

Frog Microcontroller Trainer (FMT): Analog Write
Length of Time: 55 Minutes
Recommended for Intermediate

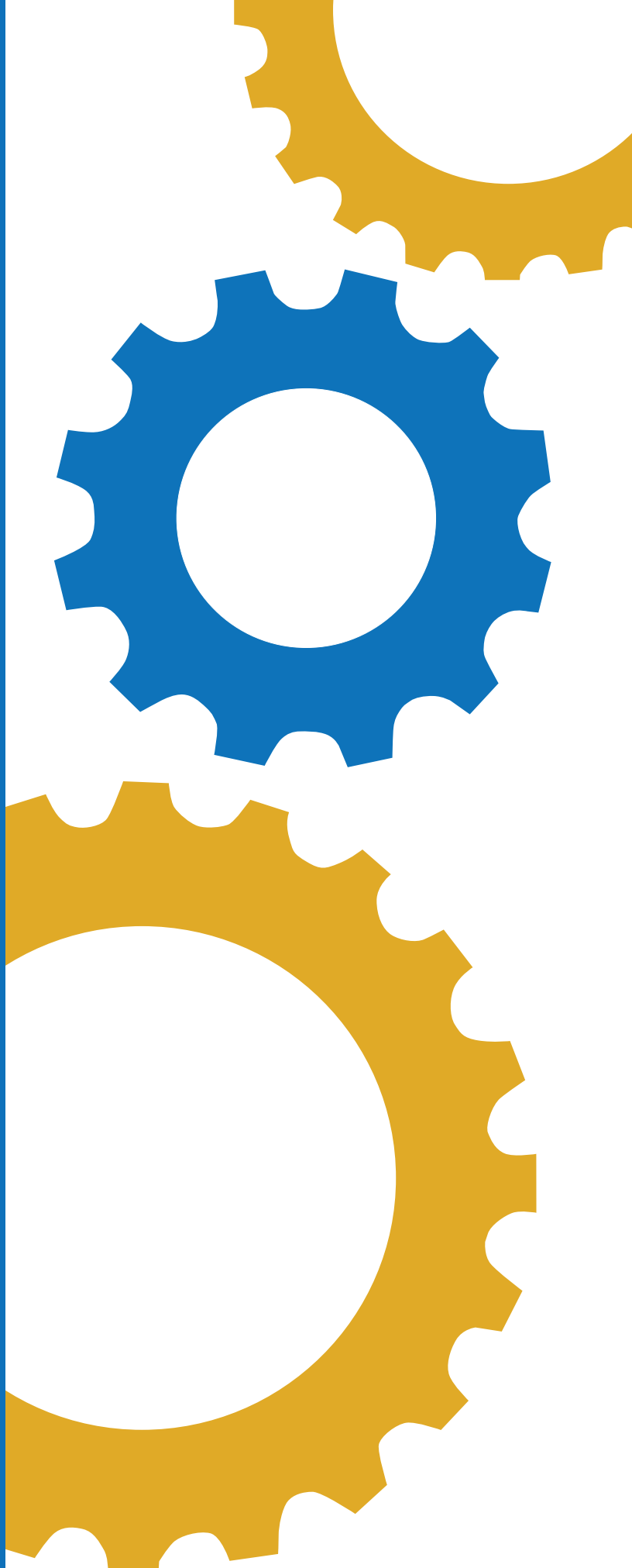
Overview

- The FMT is capable of reading analog and digital inputs. In this lesson, we will read a digital input from a button.

Objectives:

Students will be able to:

- Copy the 50% duty cycle analog write function
- Test sketch on FMT
- Calculate 30% duty cycle and write the corresponding code
- Calculate pulse width modulation (PMW)
- Test the new sketch that the student generated





Indiana standards for Principles of Computing:

- 7183.D2.2- Apply basic logical structures, file handling, matrices, and arrays to program algorithms.

Indiana STEM Integration Standards:

- HS.DM.1- Use multiple systems of measurement (i.e., standard and metric) and data sets (e.g., plots, tables, graphs, charts) defined in course-level content standards to analyze real-world scenarios and the mathematical relationships represented by the data.
- HS.AM.2- Create advanced models (e.g., mathematical models, computer simulations) to represent and explain natural and designed systems, defined in course-level content standards.

CSTA:

- 1A-CS-02 Use appropriate terminology in identifying and describing the function of common physical components of computer systems.
- 1B-AP-12 Modify, remix or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.
- 1B-AP-15 Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.
- 2-AP-16 Incorporate existing code, media, and libraries into original programs, and give attribution.

Materials:

- [Frog Microcontroller Trainer](#) from 1st Maker Space
- USB Power Cord
- Arduino IDE
- PC or Mac
- Chromebooks if using Arduino Create for Education App
- Introduction to Microcontroller Trainer Lesson (if needed as a resource)

Preparation:

- Ensure students have a compatible device and have downloaded the Arduino IDE software on device.
- Ensure all students know how to properly connect the FMT to the device via the USB cable.

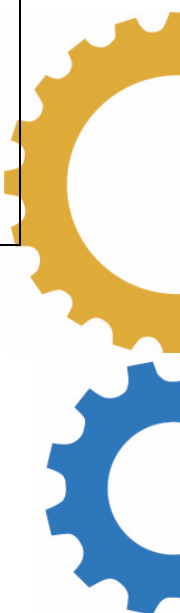
Background Information:

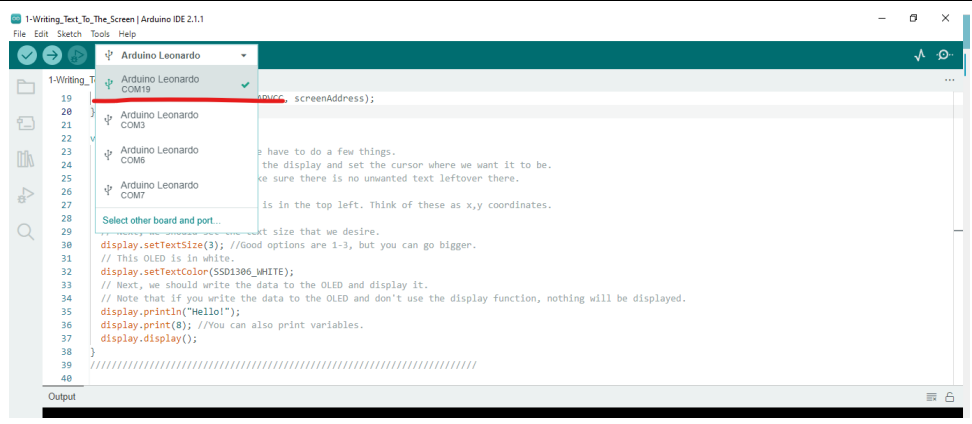
In this lesson students will learn about a new function called analogWrite. This function is useful for setting the brightness of LEDs. It can also be used to set the speed of motors, servo motor position, generating audio tones, and more.

Lesson Elements



<p>Introduction to duty cycles, square waves and PWM– 20 minutes</p>	<p>Provide each student a copy of the student handout. Go over what duty cycles and square waves are and why they're important. Introduce pulse width modulation (PWM) and work through the provided example (50% and 90%) with the class.</p>
<p>Access 1MS Code – 15 minutes</p>	<p>Visit the 1st Maker Space website and access the library of code developed specifically for the FMT. Here, you will find a section with various code (called projects) written for each major component of the trainer. The teacher will locate the sketch by clicking on 1.03- Analog Write.</p> <p>← Back to Project List</p> <h3>Project 1.03 Analog Write</h3> <p>In this project you will learn about a new function called analogWrite. This function is useful for setting the brightness of LEDs. It can also be used to set the speed of motors, servo motor position, generating audio tones, and more.</p> <p>The MC Trainer can only produce values of 0v and 5v but what happens when you need 2.5v? Simply put, the MC trainer cannot produce 2.5v. Instead, what it can do is switch a pin between 0v and 5v very quickly and produce an average voltage of 2.5v. This process is known as PWM.</p> <p>It is important to note that this can only be done with certain pins. In our case, LED1 and LED2 can be used to do PWM.</p> <p>This also introduces the Idea of duty cycle. Duty cycle is the ratio of how long a signal is on VS how long a signal is off. In this case the duty cycle would be 50%.</p> <p>Here, one can find the sketch to share with students. Students can simply copy the code by selecting the blue copy button.</p> <p>Project Code:</p> <div style="border: 1px dashed gray; padding: 10px;"> <pre> //////////////////////////////////// // 1.03 - Analog Write byte LED1 = 13; void setup() { pinMode(LED1, OUTPUT); analogWrite(LED1, 127); } void loop() { } ////////////////////////////////////</pre> <div style="text-align: right;">Copy</div> </div>
<p>Connect the Trainer and Verify the Sketch– 10 minutes</p>	<p>Plug the Frog Microcontroller Trainer into your computer's USB port. Using the dropdown box near the top, select the Arduino Leonardo and appropriate COM port. Note: Mac computers may automatically select the port.</p>

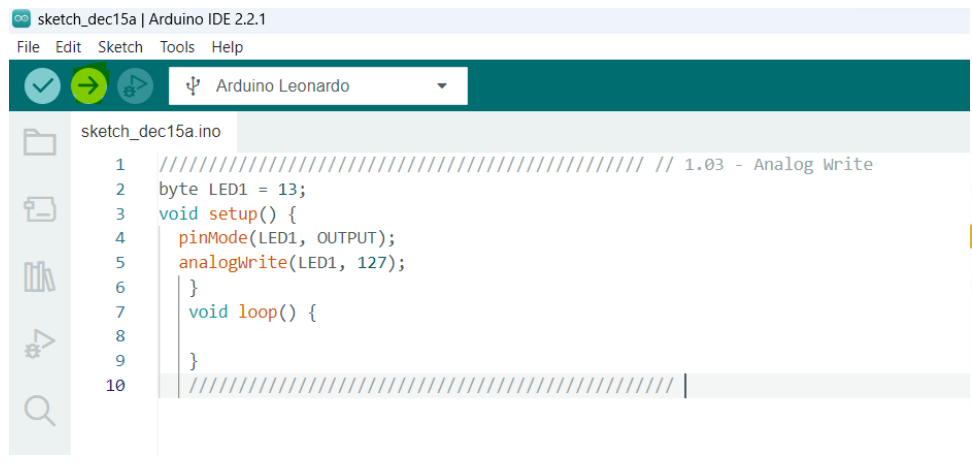




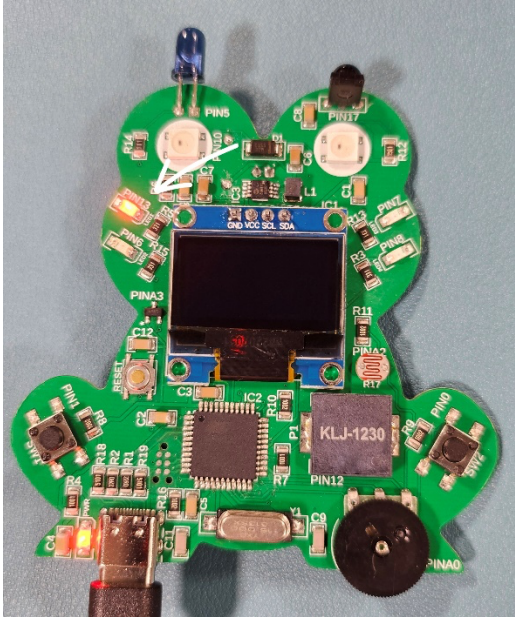
Making sure the USB cable is connecting the FMT to the computer, verify the sketch by clicking on the check mark in the upper left-hand corner.



Once verified, upload the sketch to the trainer by clicking the right-facing arrow. It will turn yellow.





	<p>To understand if the sketch is working correctly, the LED 1/Pin 13 should now be illuminated.</p> 
<p>Perform Calculations and Modify the Sketch – 20 Minutes</p>	<p>Once students have successfully run the sketch, they will now calculate the 30% duty cycle square wave, produce a square wave using the slider in the online lesson and modify the sketch to reflect their calculation. If done correctly, students should illuminate the same LED 1/Pin 13 but it should appear dimmer than before.</p> <p><u>Solution:</u></p> <p>30% of 5V= 1.5V 1.5V/.0196=77</p>





The screenshot displays an Arduino IDE window titled "sketch_dec15a | Arduino IDE 2.2.1". The interface includes a menu bar (File, Edit, Sketch, Tools, Help) and a toolbar with icons for check, run, and upload. The selected board is "Arduino Leonardo". The code editor shows the following code in "sketch_dec15a.ino":

```
1 ////////////////////////////////////////////////// // 1.03 - Analog Write
2 byte LED1 = 13;
3 void setup() {
4   pinMode(LED1, OUTPUT);
5   analogWrite(LED1, 77);
6 }
7 void loop() {
8 }
9 }
10 //////////////////////////////////////
```

Below the code editor, there is a control panel with two sliders: "Duty Cycle (%)" set to 30% and "Number of Cycles" set to 5. A graph shows a square wave signal between 0V and 5V. Below the graph is a small green LED indicator labeled "LED Brightness:".

Career Exploration:

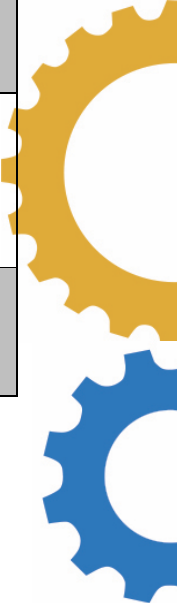
- A good resource is the [IDOE Career Explorer database](#).
- Another good resource for career information is the Bureau of Labor Statistics

Practical Application:

The MCU Trainer can only produce voltages of 0V and 5V. So, to produce lower voltages, we use PWM. This can be used to dim LEDs, drive motors, send signals and more.

What's Next?:

- FMT: Project 3.02 and 3.03- AND/OR Logic





Additional Resources:

- 1st Maker Space Frog Microcontroller Library
- 1st Maker Space Learn Arduino with the Frog Microcontroller

Performance Assessment/Check for Understanding:

- Was the student able to successfully load the sketch to the FMT?
- Could the student calculate PWM?
- Was the student able to modify the sketch to dim the LED?