LAB 5: MATLAB GUI & ARDUINO PROGRAMMING

BROUGHT TO YOU BY....

Henry Liu, Nathaniel Seekins, & Alexis Yu

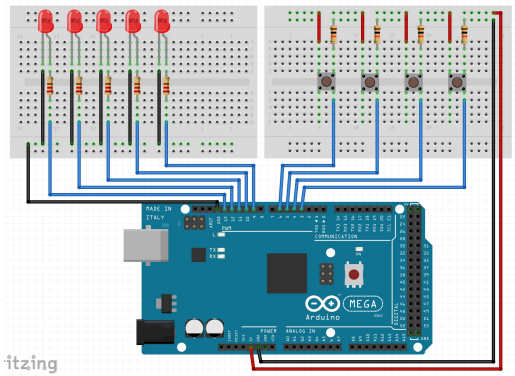
LAB 5 OVERVIEW / OBJECTIVE (Nate)

- Creating a whack-a-mole game
- Using experience and understandings of Arduino and MATLAB GUI
- Building circuits from fritzing and circuit diagrams
- Using OnShape to 3D prototype this game

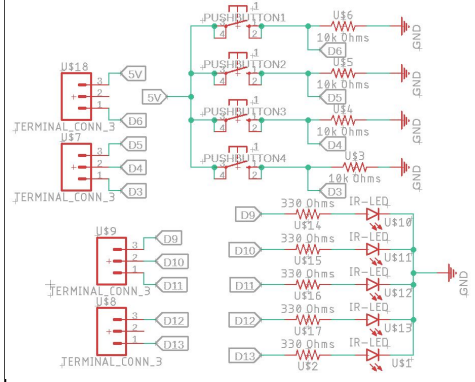
CHALLENGE #1 (Nate)

- Create a circuit that contains four buttons and 5 led lights
- Have an led light up when the buttons have been pressed twice (any of them or the same one)

CHALLENGE #1 (Nate)



- Fritzing diagram



- Circuit diagram

Useful resources:

Push Button Reference

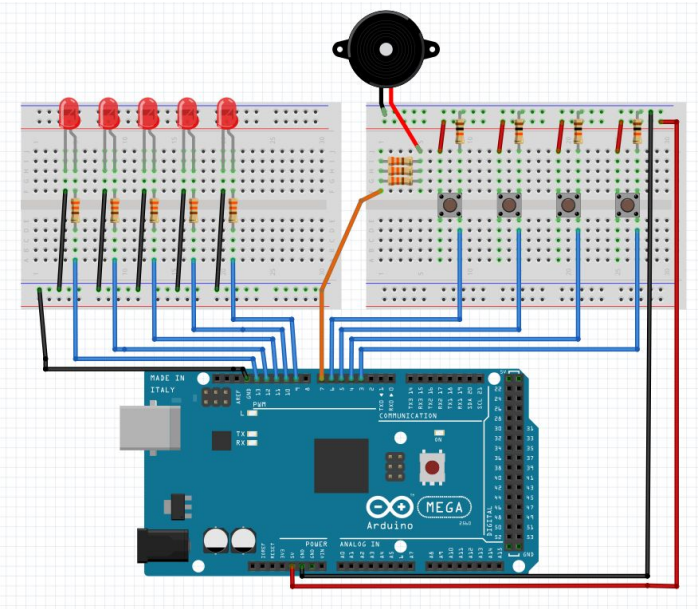
LED Reference

Checking for a button press in Arduino

CHALLENGE #2 (Henry)

- Incorporate a piezo buzzer into the circuit built from Challenge #1
- Have each button press emit a sound
- Play a tune when all of the LEDs are illuminated

CHALLENGE #2 (Henry)



- Fritzing diagram for circuit to be constructed in Challenge #2
- An extension of Challenge #1
- Adds a piezo buzzer
- Introduces parallels series and mathematics of total resistance

CHALLENGE #2: Coding a Piezo Buzzer Basics (Henry)

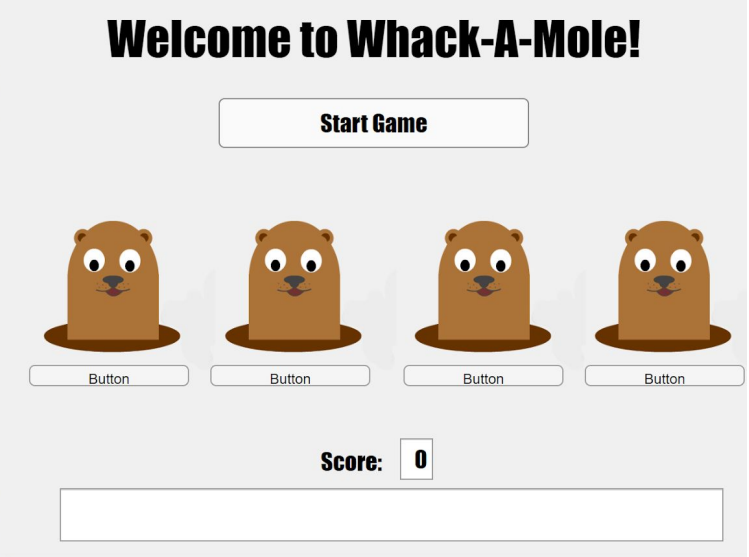
```
//A sketch to demonstrate the tone() function

//Specify digital pin on the Arduino that the positive lead of piezo buzzer is attached.
int piezoPin = 8;

void setup() {

}//close setup

void loop() {

  /*Tone needs 2 arguments, but can take three
    1) Pin#
    2) Frequency - this is in hertz (cycles per second) which determines the pitch of the noise made
    3) Duration - how long teh tone plays
  */
  tone(piezoPin, 1000, 500);

  //tone(piezoPin, 1000, 500);
  //delay(1000);

}
```

- Initialize piezo buzzer to a pin
- Set a frequency (Hz) for the sound pitch
- Set a time duration in milliseconds to emit sound
- Difference between delay and sound duration is the pause time in milliseconds

CHALLENGE #3 (Alexis)

- Develop front end of MATLAB GUI Whack-A-Mole game
- Must contain start button, score counter, four non-visible buttons, four visible buttons, an edit text box, and mole graphics

Challenge #3 (Alexis)



- Example GUI
- We will develop GUI into game in challenge #4
- Lay Out Apps in App Designer View
- Create a Simple App Using GUIDE

CHALLENGE #4 (Alexis)

- Program Whack-a-Mole game using GUI
- Players must click on a mole within a certain time period to receive point
- Will implement callback functions and counters

References:

Startup Tasks and Input Arguments in App Designer

Display Graphics in App Designer

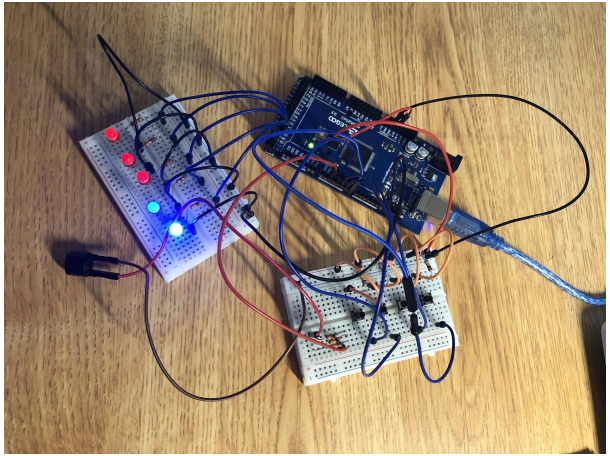
CHALLENGE #4 (Alexis)

The user will need to write:

- 4 callbacks for the buttons underneath the moles
- A callback for the start button
- A function that will randomly display a mole
- Add 1 point to the score counter whenever a mole is hit and subtract one whenever a button is wrongly hit
- Have the text say congratulations once the score reaches 10 points

CHALLENGE #5 (Nate)

- Arduino and MATLAB GUI will be integrated to create Whack-a-Mole game
- Create an Arduino controller using the circuit components from Challenge #1



- Read Streaming Data from Arduino Using Serial Port Communication

CHALLENGE #5 (Nate)

We will be using several new functions that will let MATLAB control the arduino including:

- readDigitalPin
- playTone
- writePWMVoltage

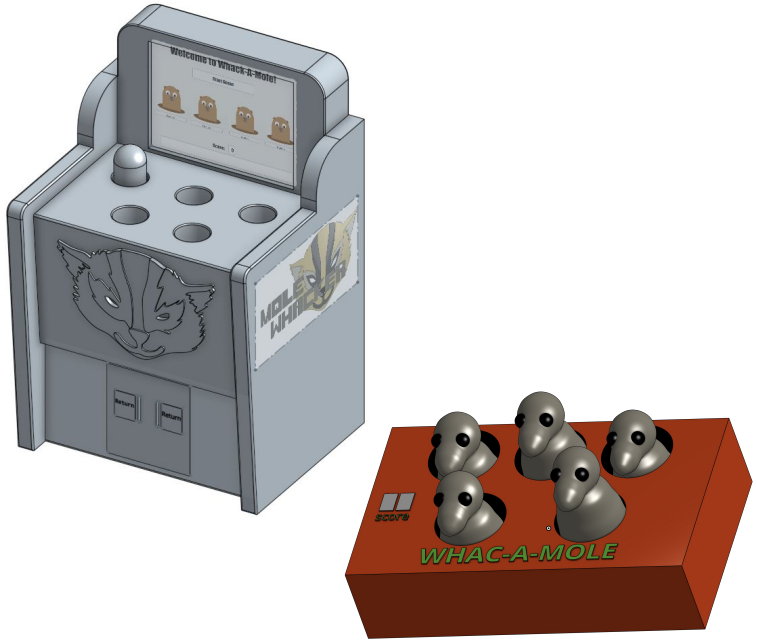
Link to Video Explaining Functions:

https://www.mathworks.com/videos/arduino-and-matlab-reading-inputs-and-writing-outputs-106502.html

CHALLENGE #6 (Henry)

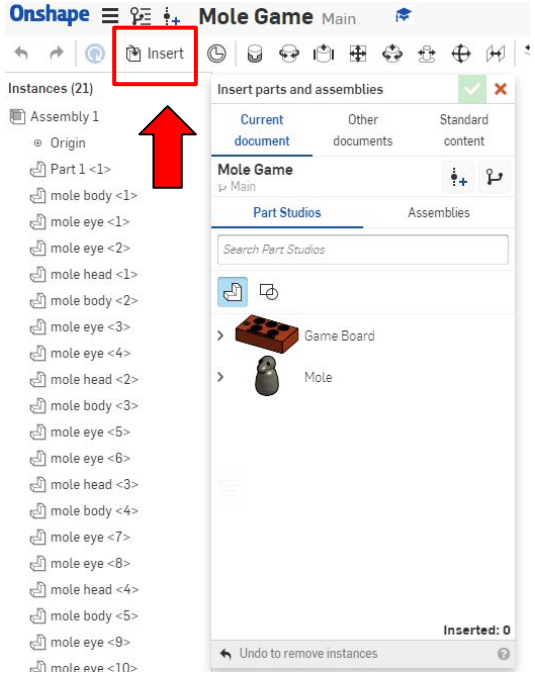
- Create a 3D prototype for what this game may look like at an arcade using OnShape

CHALLENGE #6 (Henry)



- Examples of what we will be creating
- Idea is to create individual parts
- Put them together using OnShape's assembly feature

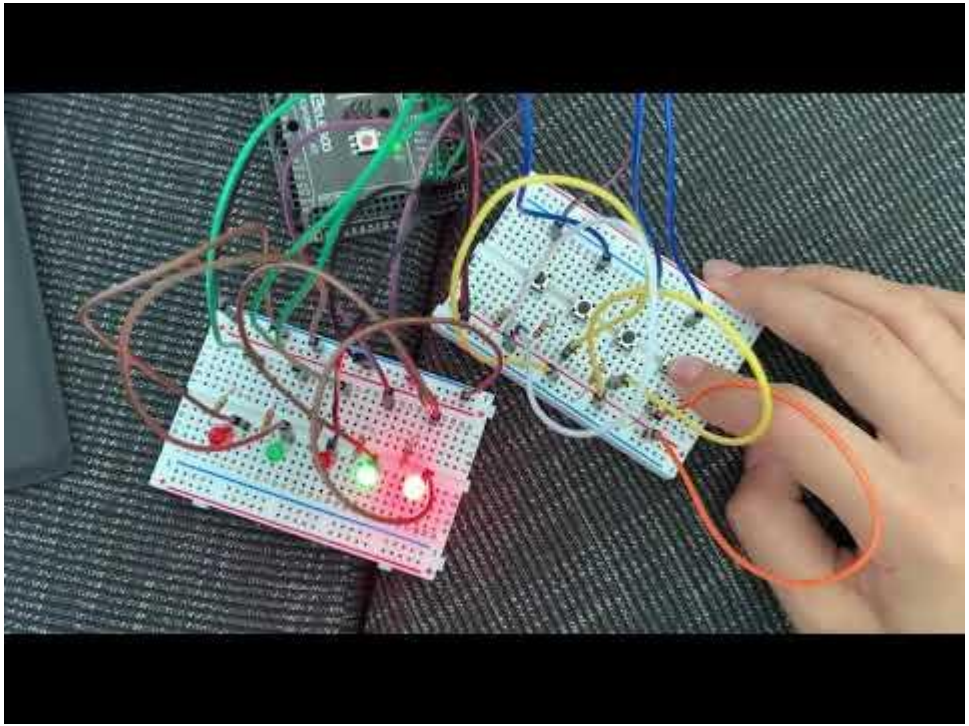
CHALLENGE #6 (Henry)



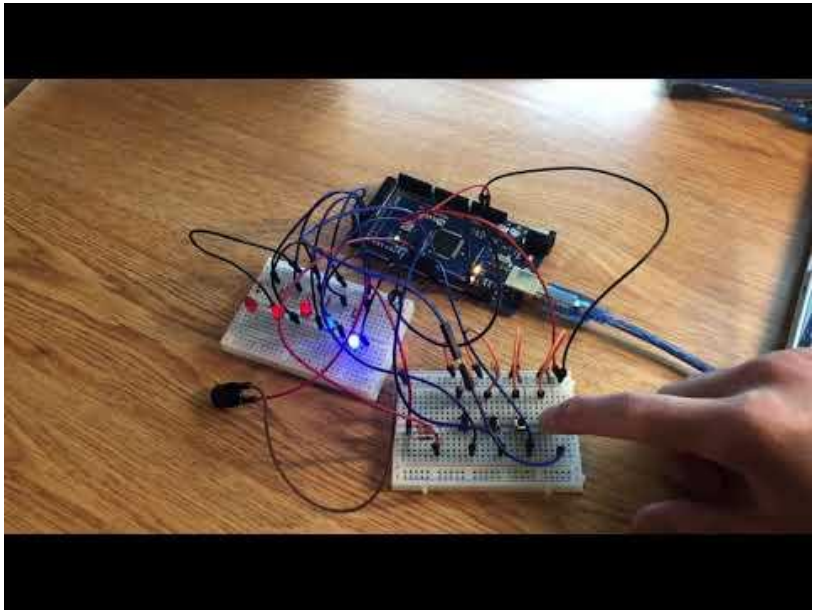
- Navigate to assembly tab on the bottom of the screen
- Use insert to include the parts you have created
- Drag the parts into place with transform capability

LAB 5 RECAP

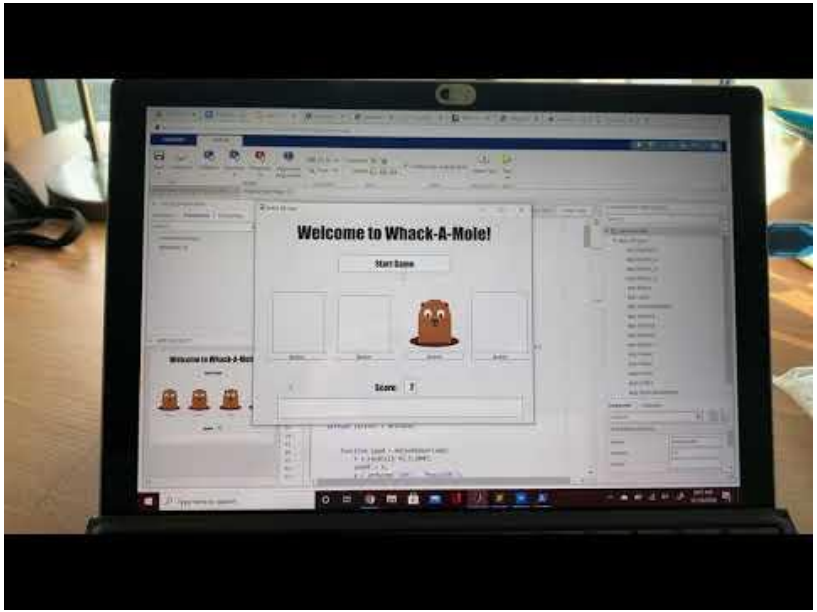
CHALLENGE #1 DEMO (Alexis)



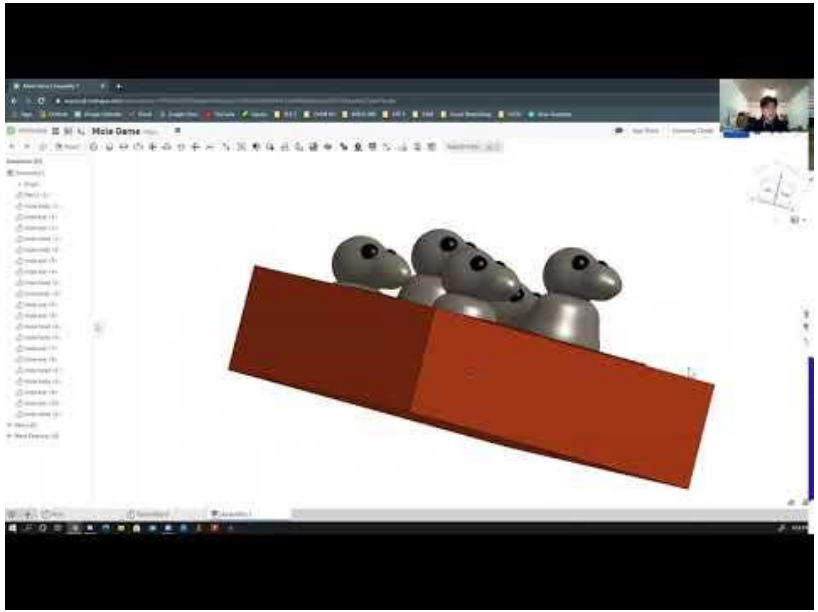
CHALLENGE #2 DEMO (Henry)



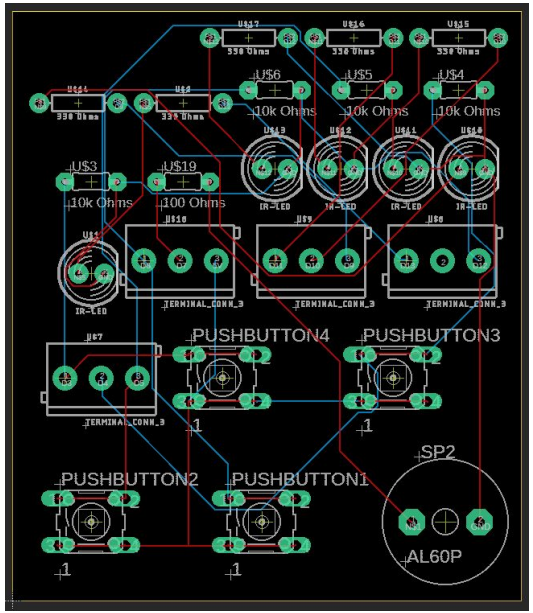
CHALLENGE #3, #4, & #5 DEMO (Alexis)



CHALLENGE #6 DEMO (Henry)



PRINTED CIRCUIT BOARD (PCB) (Henry)



- Example of what our game circuitry would look like if we were to manufacture a PCB for it

RESOURCES

Challenge #1
For more information regarding how the code works and runs, please visit:
https://medium.com/arduino-playground/checking-for-a-button-press-in-arduino-7681cbb7bde7

Challenge#2
For more information regarding how a piezo buzzer makes noise, wired, and coded, please visit the following links:
https://create.arduino.cc/projecthub/SURYATEJA/use-a-buzzer-module-piezo-speaker-using-arduino-uno-89df45
Watching the following video may be especially helpful
https://www.programmingelectronics.com/an-easy-way-to-make-noise-with-arduino-using-tone/

For more on calculating the total resistance in a parallel circuit, please visit:
https://www.goodscience.com.au/year-9-physics/electrical-resistance-and-ohms-law/5-calculating-total-resistance-in-a-circuit/

For an explanation of how a parallel circuit works, please visit:
https://learn.sparkfun.com/tutorials/series-and-parallel-circuits/all

Challenge#3
Use this link to understand how to set up an app in GUI:
https://www.mathworks.com/help/matlab/creating_guis/about-the-simple-guide-gui-example.html

RESOURCES

Challenge #4
Startup Tasks and Input Arguments in App Designer:
https://www.mathworks.com/help/matlab/creating_guis/app-designer-startup-function.html
Displaying Graphics in App Designer:
https://www.mathworks.com/help/matlab/creating_guis/graphics-support-in-app-designer.html

Challenge#5
Read Streaming Data from Arduino Using Serial Port Connection:
https://www.mathworks.com/help/matlab/import_export/read-streaming-data-from-arduino.html
Connect to and control Arduino inputs and outputs from MATLAB:
https://www.mathworks.com/hardware-support/arduino-matlab.html
Arduino interface from MATLAB:
https://www.allaboutcircuits.com/projects/arduino-interface-with-matlab/
Video on MATLAB functions:
https://www.mathworks.com/videos/arduino-and-matlab-reading-inputs-and-writing-outputs-106502.html

Challenge#6
Basics of OnShape:
https://www.youtube.com/watch?v=kGLZcrLkmHM
Creating an assembly in OnShape:
https://www.youtube.com/watch?v=YgxjGmrjDRU