# Whac-A-Mole

Zechen Zhang and Xu Cai
Boston University
zzc0423@bu.edu, xuc2019@bu.edu

May 2, 2022

**Abstract**

***We have all played this classic arcade game when we were little: The Whac a Mole. The game reminds me of my happy childhood, and hence our final project for PY371 eLab is to realize this game mechanism using Arduino and a SNES controller. We hope this self-explanatory and fun project would remind players their joyful childhood as well. This paper gives an insight into the game mechanism, parts, block diagram, setups, and codes we worked for this project in the past five weeks.***

## I. Introduction

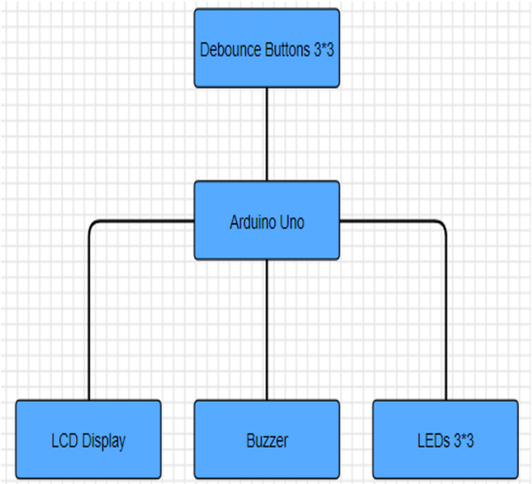
The mechanism of the game is very straight forward. The player needs to hit the correct buttons signaled by the LEDs, and whoever hits the most number of targets wins. The targets are represented by 8 LEDs mounted on the breadboard, and the player needs to tap on the specific button on the controller at correct time frame. The layout of LEDs resembles the layout of buttons on the SNES controller so that the player could easily identify which button they should strike. We also created a functional DIY controller using debounce buttons and a separate piece of breadboard, just for more originality. Incorrect button tapping or bad timing, too fast or too slow, will be regarded as a miss that counts as zero. Each valid tapping adds 1 point to the player's total score. The game should come with 2 different sounds that matches the key striking from players. One high pitch tone for a successful score, another low pitch tone for a failure hit. Each game lasts for 1 minute precisely. After each game, the points the player get for hitting the correct key at the right time will be calculated and displayed on the screen. Record high scores will remain on the display, waiting to be challenged by new players.

## II. Parts

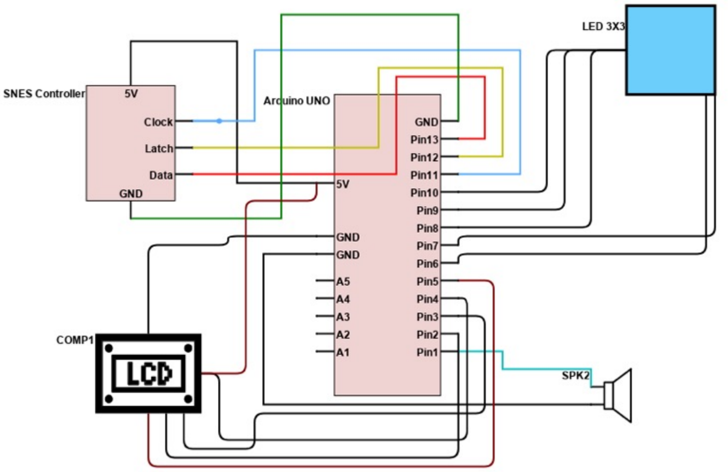
The project contains the following hardware parts:

- An Arduino Uno micro-controller
- A SNES controller
- An Arduino LCD Display
- Two buzzers
- Eight LEDs
- Breadboards
- 8 debounce buttons

## III. Block Diagram



## IV. Circuit Diagram



## V. Implementation and coding

The coding for Arduino requires detecting inputs from players, determining whether the inputs were valid or not, calculating player's scores, outputting scores to the display, and generating specific sounds through the buzzers. One of the most difficult parts of our project was to decode the signal input from a SNES controller to voltage output that could be recognized by the Arduino. The signal is in 12-bit serial binary format, and each button on the controller actually represents a position for 0 among other 11 positions for 1. The experience of creating a basic DAC converter in lab2 offered great help, and we were glad to be able to implement what we learned in eLabs onto our project. Sometimes the SNES controller fails to perform as we expected, suggesting that there's still more debugging works that need to be done. Hence, we also prepared a plan B: a DIY controller using debounce buttons, and to be honest, the coding and testing for the DIY controller was a lot simpler than the SNES controller. Integrating and debugging codes of each Arduino kit into one program that completes the Whac-a-Mole mechanism is what we are mainly working on right now, and we look forward to playing the final game once it's completed.