Dynamic Keyboard

Nigel Haran | Jeevitesh Juneja

ECE 445 | Spring 2018 | Team 20

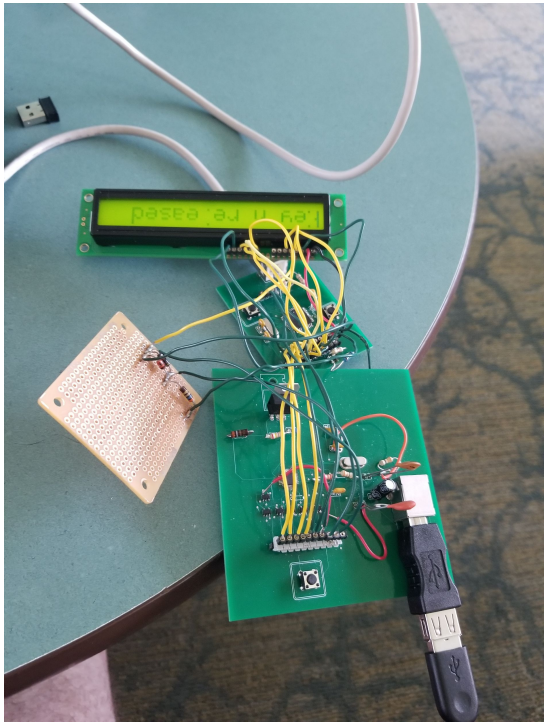
Introduction

- Keyboards are limited in their functionality and reprogrammability.
- Programmable keyboard today limit the number of keys that can be programmed and can only be utilized via specialized drives.
- Our goal:
  - Create a device that allows the user to treat any key as a programmable key
  - Allow the user to save these macros through hardware alone.
  - Utilize our device on any keyboard and computer.

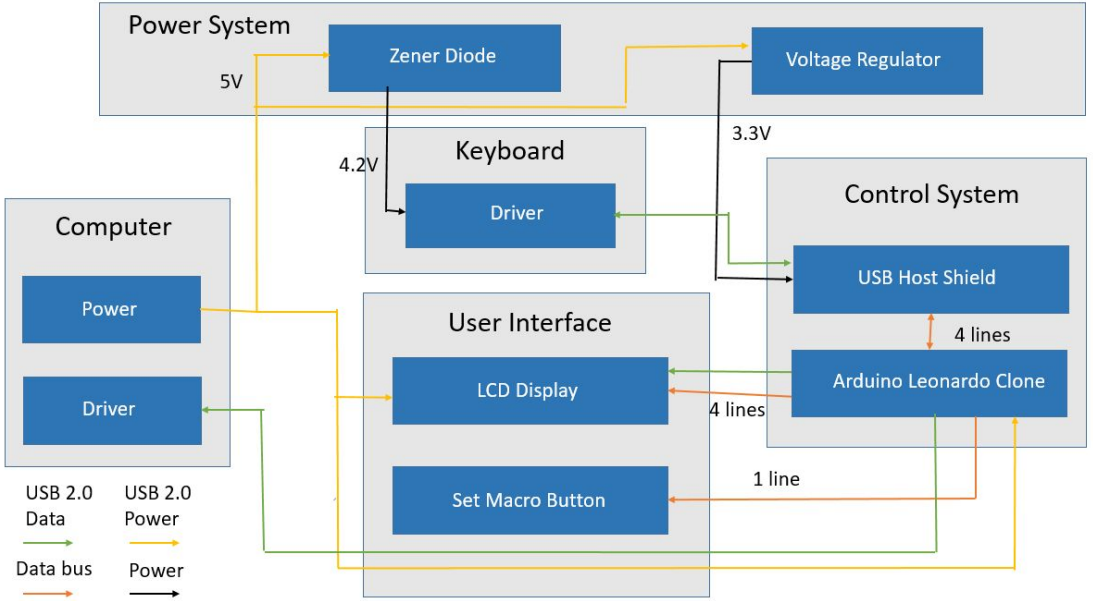
Core Features

- Provide feedback to the user through an LCD
- Write and rewrite 10 programmable macros
- Performs on any computer and any keyboard

Complete Implemeneted System



Block Diagram



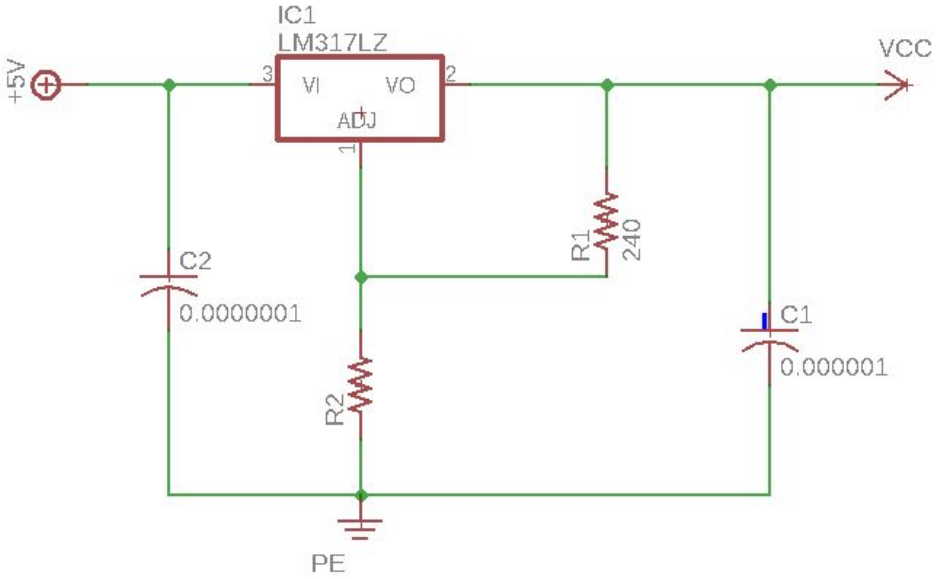
System Overview

- Voltage Regulator
- Zener Diode
- Voltage Divider
- LCD
- USB Host Shield
- Leonardo Clone
- Control Logic

Voltage Regulator

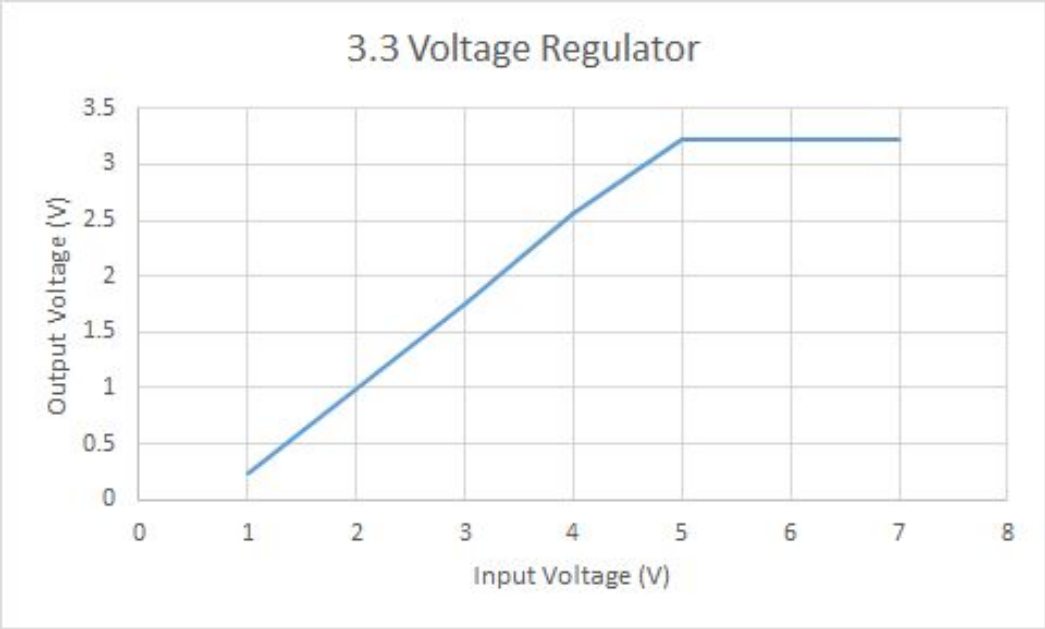
By knowing what the desired output voltage is, we determine what the value of R2 is through the equation:

Vout = Vref(1+R2/R1)+Iadj*R2



Voltage Regulator Verification

Verification of 3.3V regulator



Zener Diode

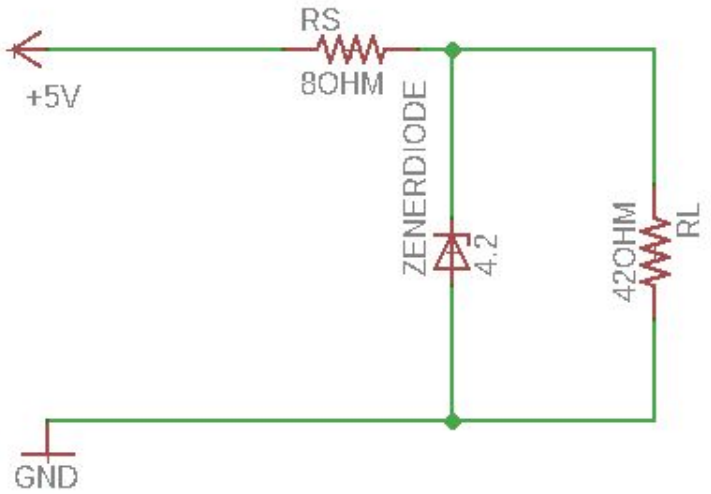
We used a Zener Diode in order to reduce the 5V input to around 4.2V with a +/- 5% error.

We had to calculate the series and parallel resistance through the equations below.

Rs = (Vs-Vz)/Iz = (5-4.2)/0.1 = 8

RL = Vz/IL = 4.2/.1 = 42

Measuring we obtained 4.05 V which was within our 5% error.



Voltage Divider

We had a voltage divider in order to act as a reference for the LCD.

The LCD needs a reference of roughly 0.5V, but after investigating we realized our LCD worked better with a 0.8V reference.

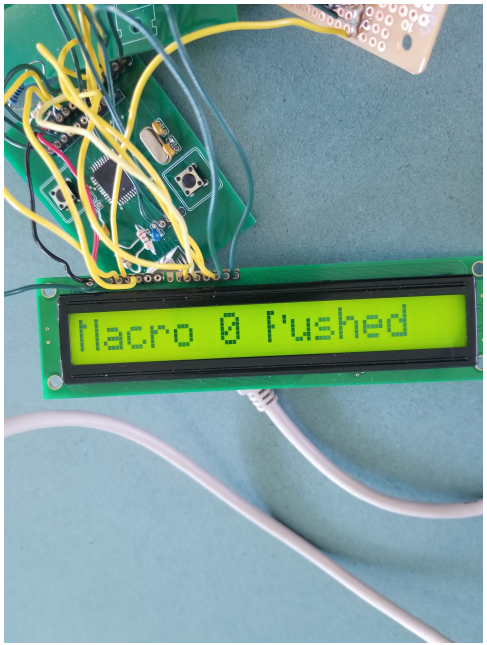
We were able to obtain this reference voltage through a voltage divider of 9.1 kΩ and 1.8 kΩ.

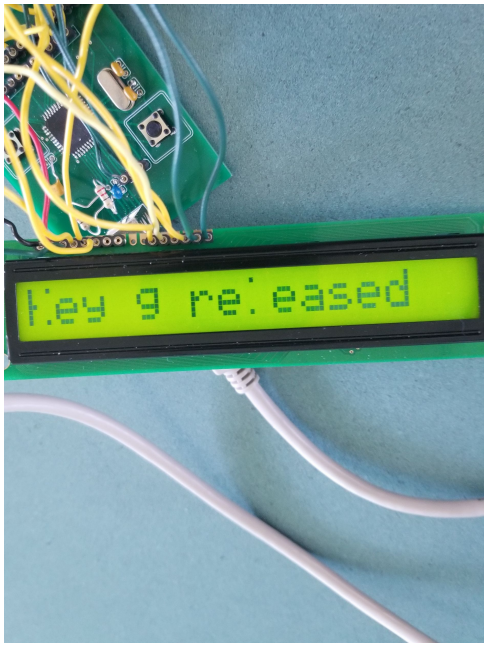
The output of the voltage divider was 0.803V which is within our 5% error.

LCD Screen (1/3)

Verification of the LCD Screen being able to display:

- Release of the keys
- Release of set macros.





LCD Screen (2/3)

Verification of LCD screen being able to display:

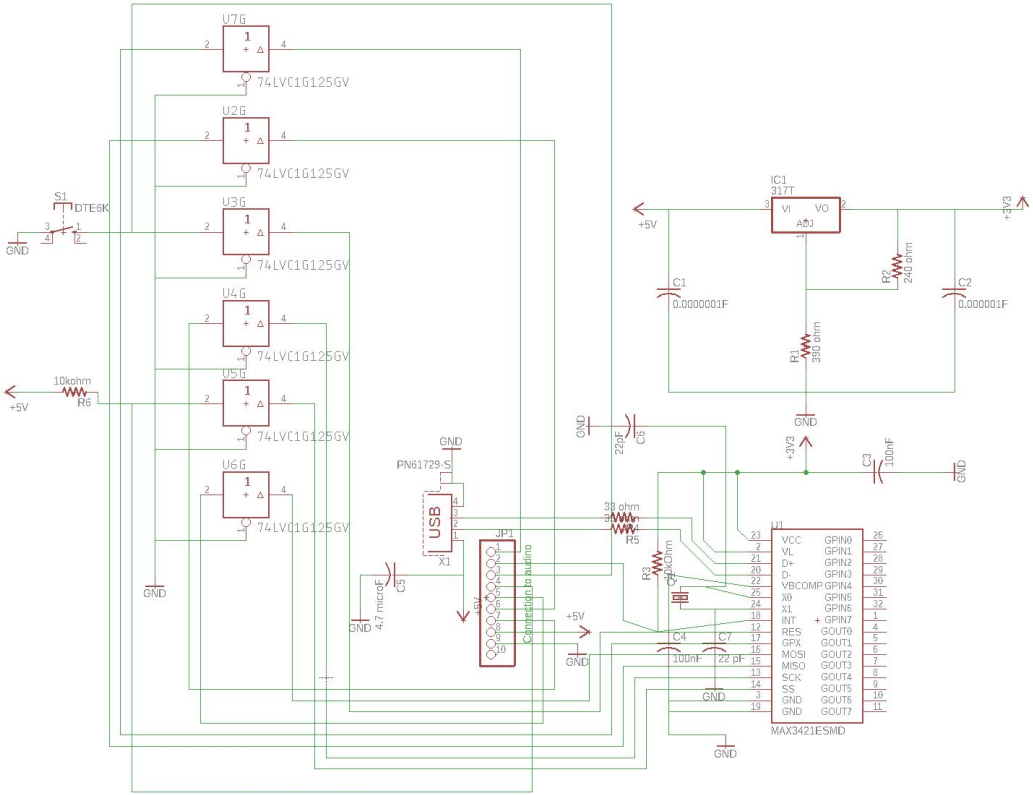
- Type Macro
- Macro Set

LCD Screen (3/3)

Verification of LCD Screen being able to display:

- Macros Reset

USB Host Shield Schematic



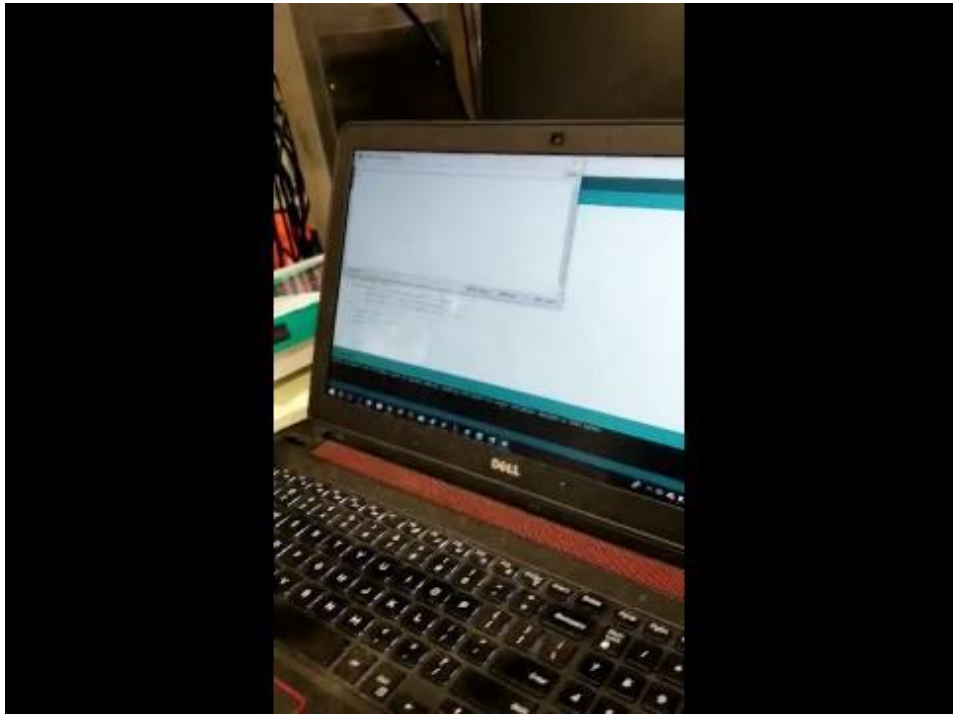
USB Host Shield

- USB protocol separates hosts and devices
- Our product needs to act as both host and device
- Host Shield provides the hardware for usb host operation

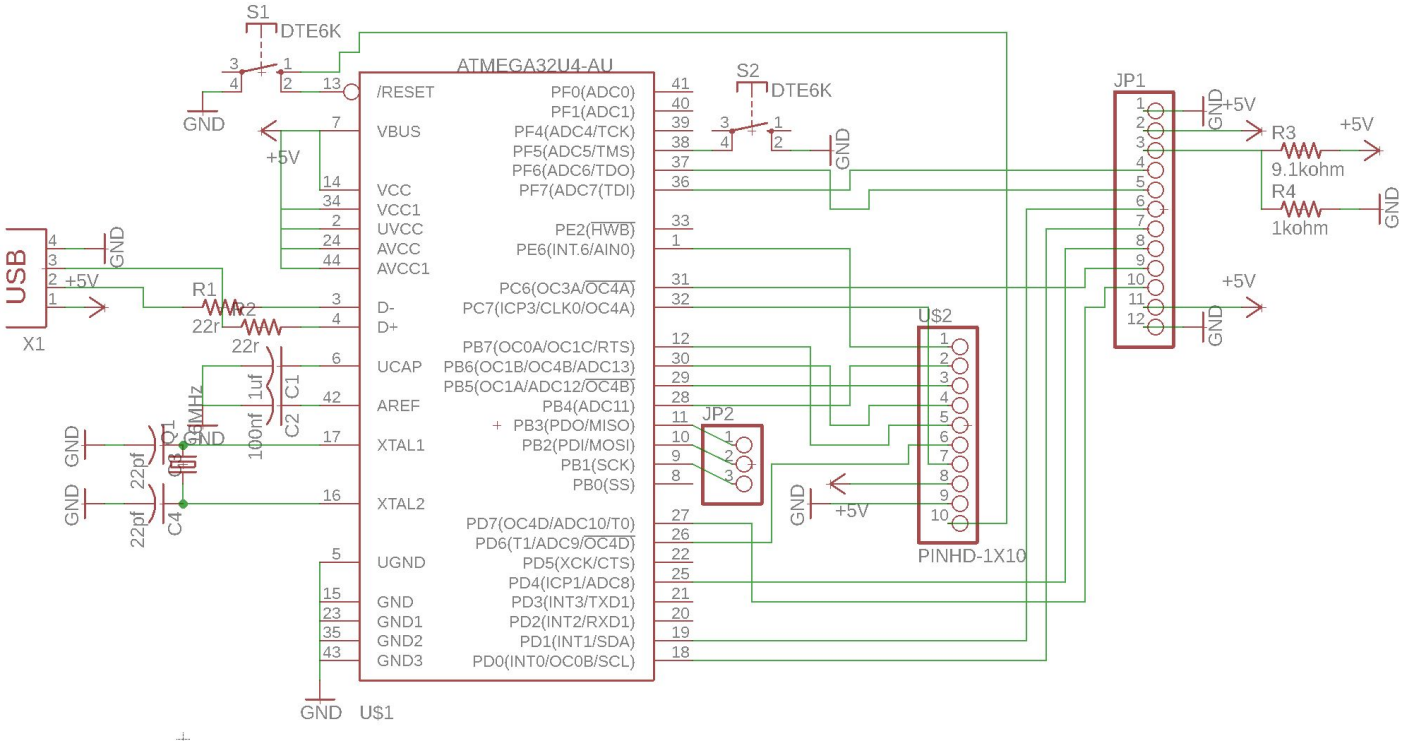
USB Host Shield Verification (1/2)

- Connected our designed USB Host Shield to a Arduino development board
- Set USB HID communication with keyboard
- Output results over serial port to the computer

USB Host Shield Verification (2/2)



Leonardo Clone



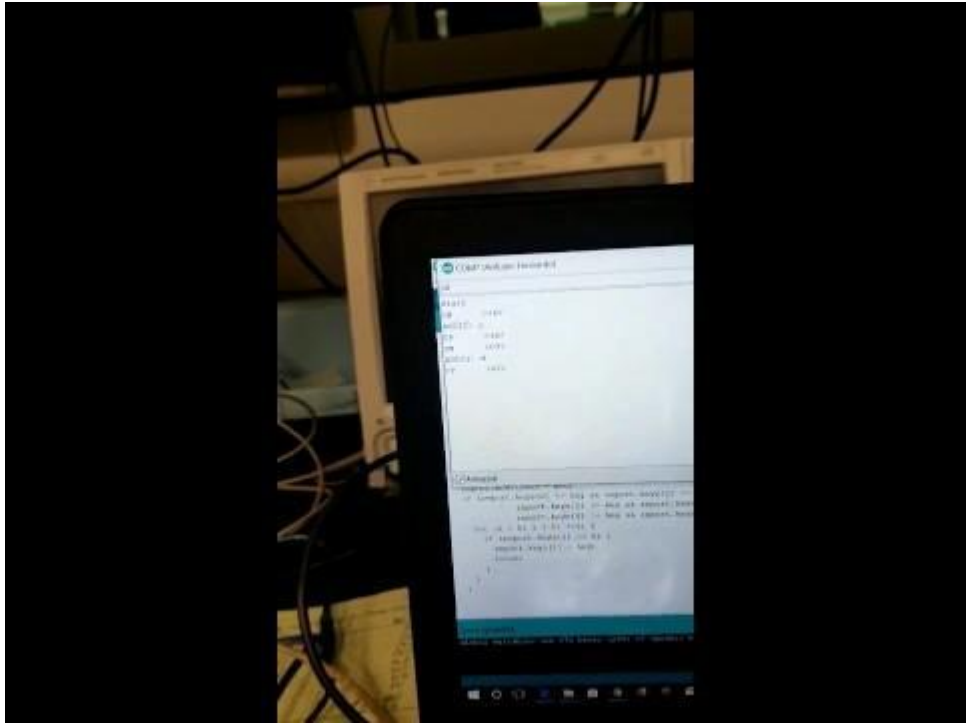
Leonardo Clone

- Acts as an USB device.
- Programmed in order to store the macros and have them set to certain keys.
- Stores the macros in EEPROM when device in not powered

Leonardo Clone Verification (1/2)

- Connected our designed Leonardo clone to a manufactured USB Host Shield
- Outputted pressed keys from the keyboard to serial port
- Have the Leonardo clone act as a USB HID device and type the keys pressed as well

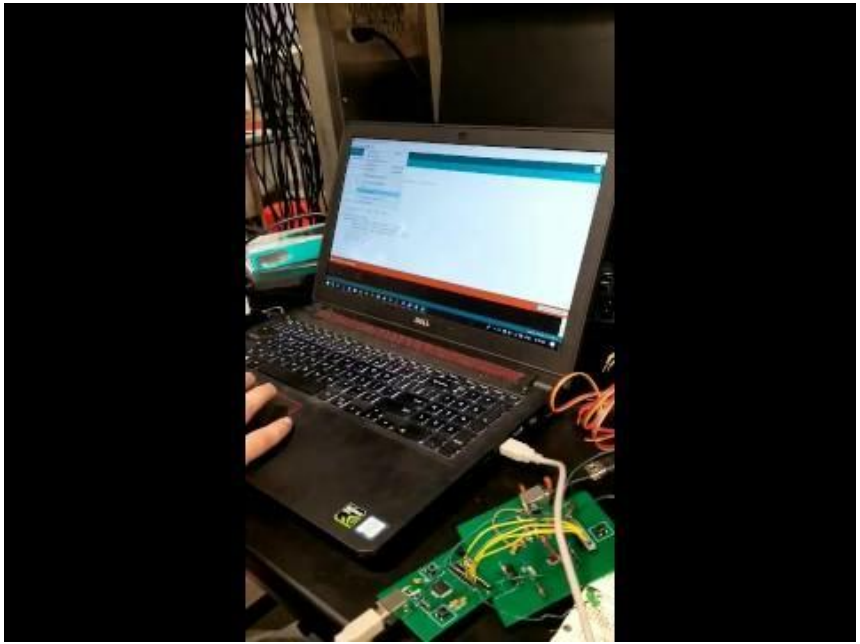
Leonardo Clone Verification (2/2)



Integrated Parts Verification (1/2)

- Connect both our Leonardo clone and USB host shield clone together.
- See if the PCB’s perform their individual operations when connected together

Integrated Parts Verification



Control Logic

- Control logic behind set macro operation
- Store macro struct EEPROM of the Leonardo clone
- Execute 10 programmable macros
- Reset Macros

Demonstration of Device

Verifies remaining functionalities:

- Have to set macros with button
- Have to have memory to store macros
- Executes all 10 macros

Moving Forwards

- Expand the functionality of the macros
- Migrate the entire circuitry to one PCB
- Design a case
- Ensure safety of LCD screen

Questions?