Magnetic Resistance System for Augmented Play of Mario Kart

PRESENTED BY: BLAKE PICKETT, SAGE MOONEYHAM, TYLER CHITTUM, BENJAMIN REED, AND RAY DURLIN

SUPERVISOR: MR. JESSE ROBERTS

Demonstration

https://youtu.be/qGHIfFxjndI

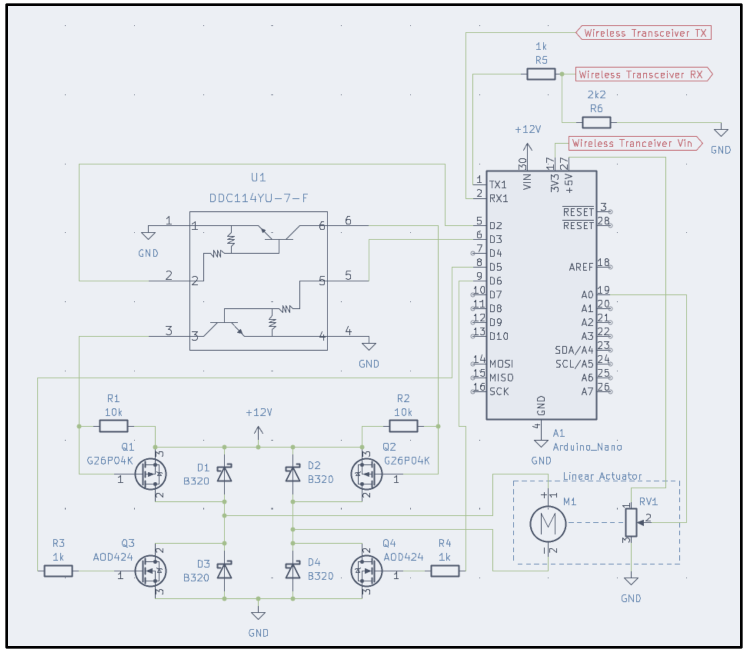
Problem

Primary Objective: Improve the Mario Kart gaming simulation platform

- Design and construct a dynamic resistance system that provides a realistic experience
- Design and construct a data recording module to be replayed on the Mario Kart simulation platform
- Convert sensors to wireless and provide new power sources
- Update wired connected sensors to wireless
- Update HDMI interface and power distribution
- Strengthen and stabilize sensor mounts

Dynamic Resistance System – Design

- Resistance felt by the user to be applied comparable an exercising spin bike
- Eddy currents created by moving two magnets closer to a rotating aluminum disk
- Modes of operation
  - Mario Kart Gaming Simulation
  - Pre-recorded course input from RRSA
- Resistance system control
  - PID controller
    - Arduino Nano
    - PID_v1 library
    - Maintains 25 % duty cycle
  - H-Bridge
    - MOSFET H-Bridge made with discrete components



Dynamic Resistance System – Design

THEORETICAL APPLICATION

$$T_d = \frac{P_d}{\theta} = \frac{\pi\sigma}{4} D^2 d B^2 R^2 \theta$$

where,

$\sigma = Conductivity\ of\ flywheel\ material\ (\Omega^{-1}m^{-1})$

$D = magnet\ diameter\ (meters)$

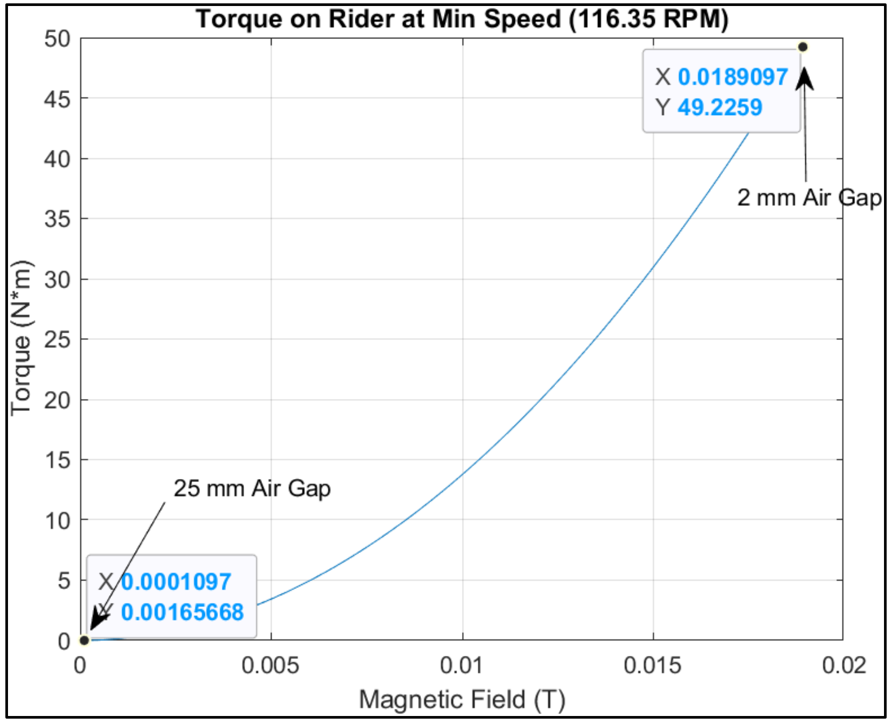
$d = disc\ (flywheel)\ thickness\ (meters)$

$B = magnet\ field\ strength\ (Tesla)$

$R = dist\ from\ flywheel\ center\ to\ magnet\ (m)$

$\theta = angular\ velocity\ \left(\frac{rads}{s}\right)$

MATHEMATICAL MODEL



Dynamic Resistance System – Design

MAGNET ANALYSIS

$$B_y = \frac{\mu_0}{4\pi} * \frac{m}{4\pi}[3\cos^2(\theta) - 1]$$

Note: Field strength in the y direction

where,

$\mu_o = magnetic\,permeability\,in\,a\,vaccum\,\left(\frac{Tm}{A}\right)$

$m\ =\ magnetic\,moment\,(Am^2)$

$r = dist.\,from\,magnet\,to\,flywheel\,(m)$

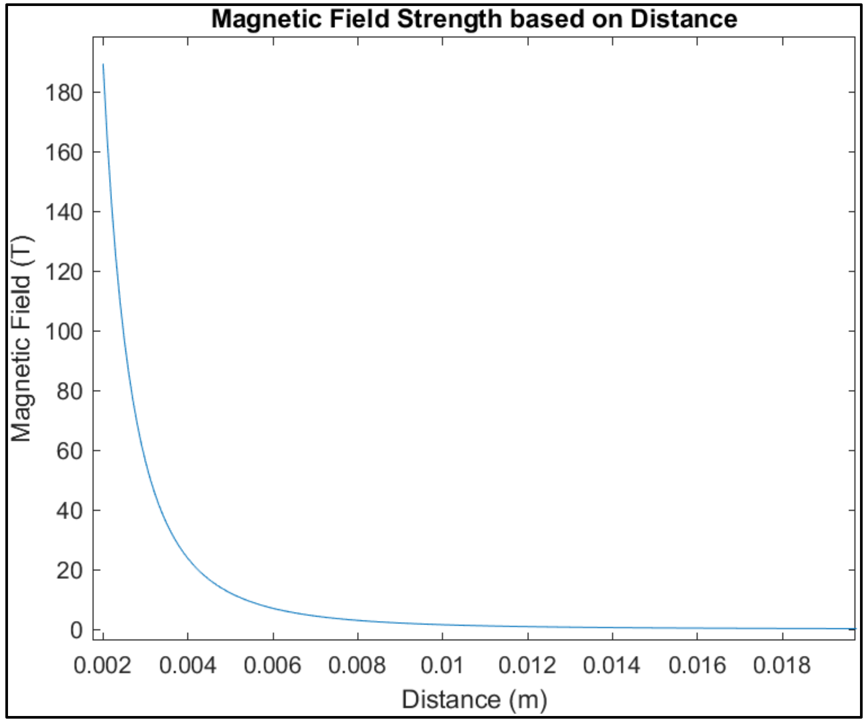
$$m = \frac{(B_r V)}{\mu_0}$$

where,

$B_r\ =\ residual\,field\ (Gauss)$

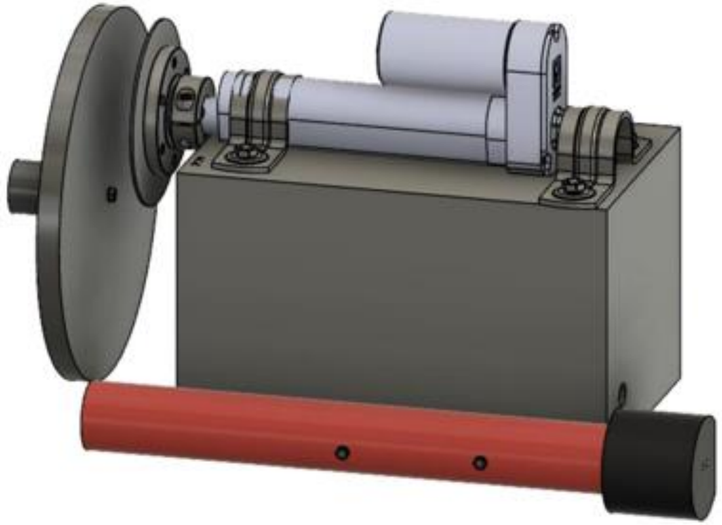
$V\ =\ magnet\,volume\ (m^3)$

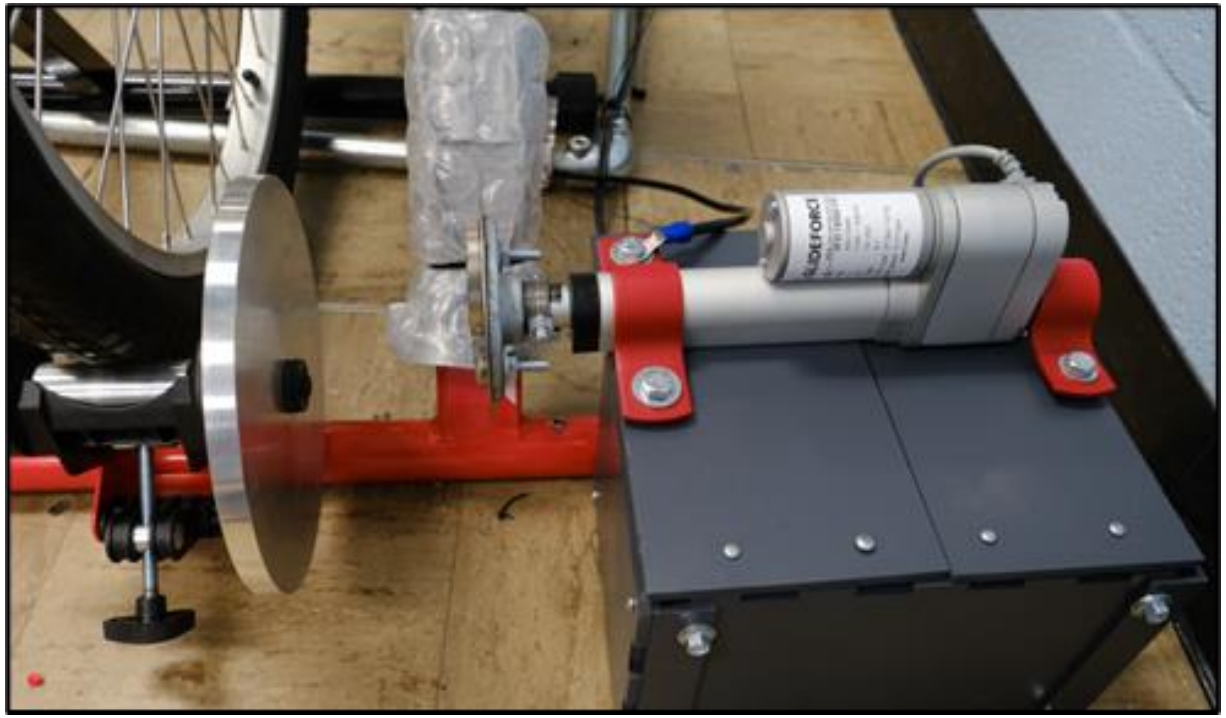
NEODYMIUM MAGNET - 2" DIA. X 1/4" THICK



Dynamic Resistance System – Design

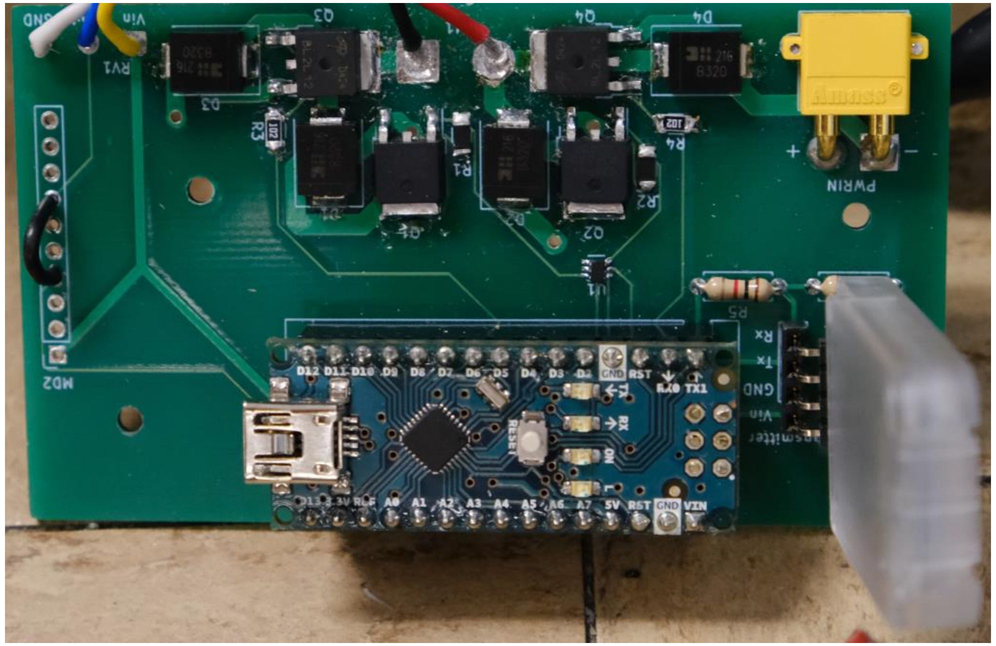
- Problems
  - Order Delays: Resistance system controller and H-bridge
  - Increased Fabrication/Build Time: fabricated flywheel for custom fit





Resistance System Controller Testing

- Some voltage ripple (8 V pk-pk maximum), but this was from supply
- Duty cycle kept at 25 %
- Not possible to test at maximum rated current due to equipment limitations
- Passed stability test



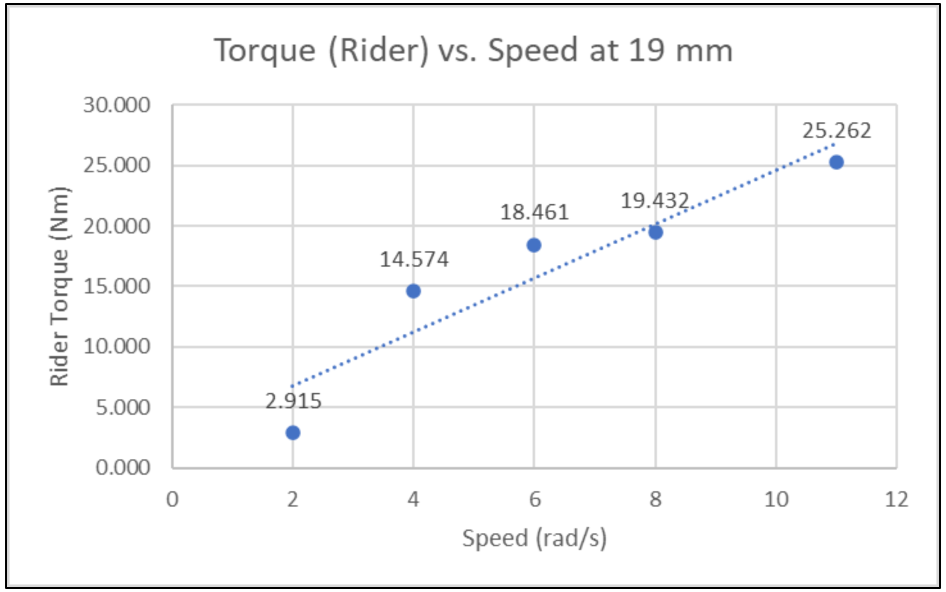
Initial Experimentation and Validation

PROOF OF CONCEPT

Resistance Measurements

- Initial experimentation validates that resistance in the form of braking torque can be provided using designed resistance system
- At a fixed distance of 19 mm
  - Data gathered proves concept of design

RESISTANCE (TORQUE) PROVIDED AT VARYING SPEEDS



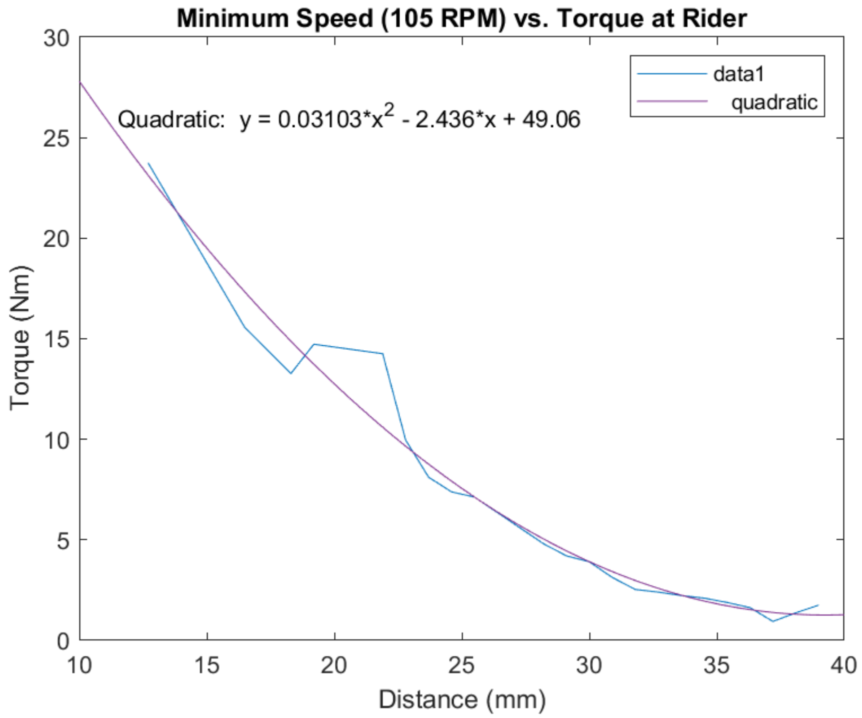
Initial Experimentation and Validation

LOW SPEED EXPERIMENTATION

Minimum speed: 105 RPM (11 rad/s)

- Distinct state: 1.434 Nm
- Max distance: 39 mm
- Min distance: 16 mm

RESISTANCE FELT BY THE RIDER AT 105 RPM

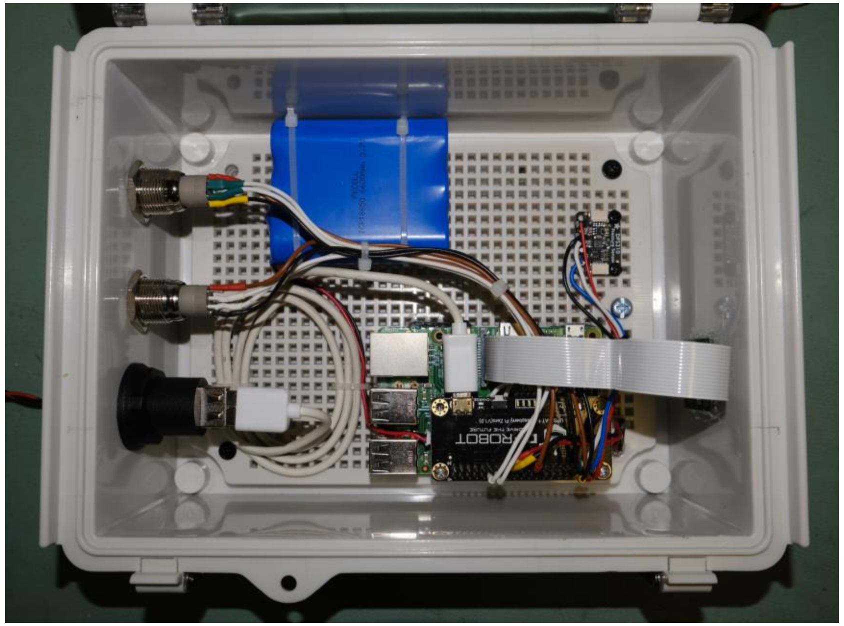


Ride Replay System Acquisition (RRSA) – Design

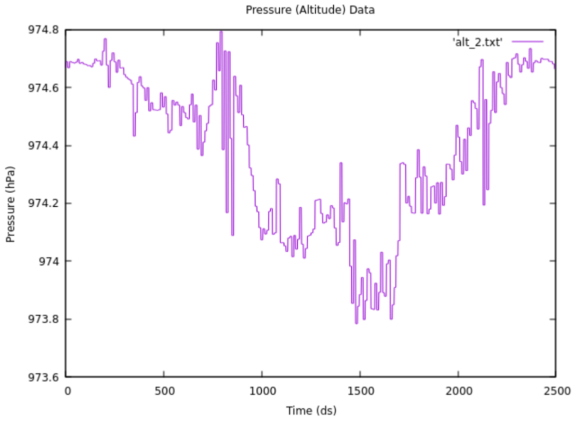
- Data Collection
  - Video
  - Elevation (via pressure sensor)
  - Speed (via reed switch and magnet)
- Data Processing
  - Uses a Raspberry Pi Model B2
  - Automatic shutdown on low battery
  - Battery life indicator LEDs
- Problems
  - Bluetooth file transfer not functional
  - Bike mounting bracket broke after testing heavy shock
  - Desync with video due to dropped frames
  - Order Delays

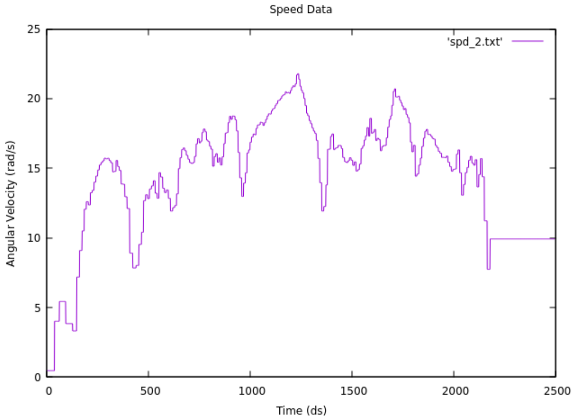
RRSA Testing

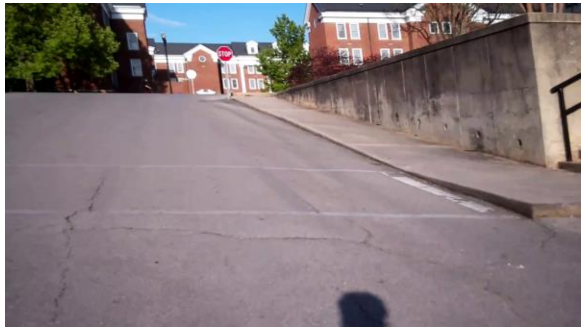
- Water resistant, but mounting bracket broke after riding off a curb
- Calculated altitude imprecision greater than 4.76 cm (design value) – failed test
- Video frame unobstructed by bike
- Video recording drops frames causing desync
- Can continuously record for at least 3 hours
- Projected recording storage capacity: 42 hours



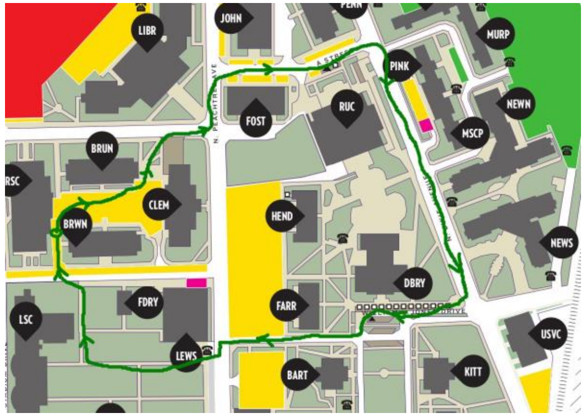
RRSA Test Example







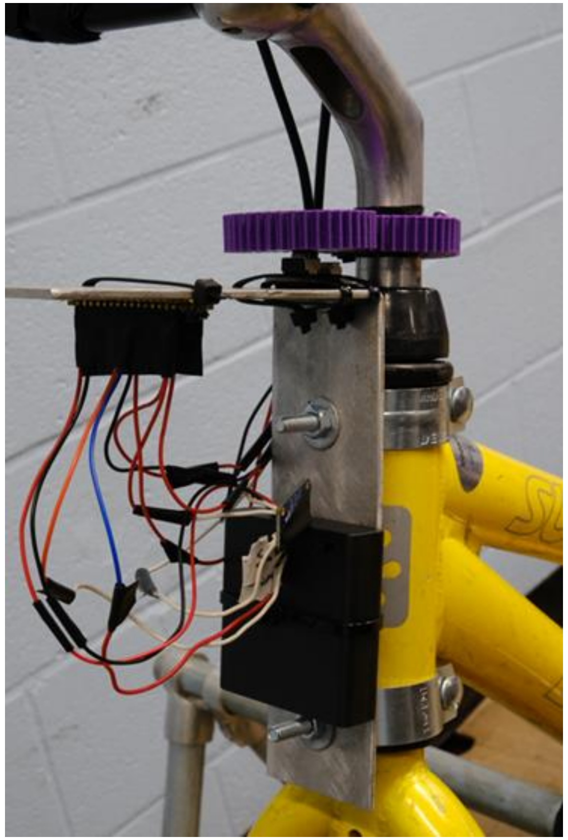
A snapshot of the collected video footage



The route followed. Map credit: TN Tech Department of Parking and Transportation

Wireless Sensors – Steering

- Design and Implementation
  - Arduino Nano BLE made the sensors able to transfer data by Bluetooth
  - Design system to enable to provide power the sensors using 4 lithium AA batteries
  - MOSFET was added in the speed sensor circuit to save power
- Problems
  - Communication: Bluetooth connection
  - Damaged Parts: Potentiometer damaged
  - Power
- Experimentation and Validation
  - Played multiple games to test the Bluetooth communication
  - Measured the voltage and current used to play a full game
  - Measured the voltage and current used while the game is not being playing



Wireless Sensors – Speed

- Design and Implementation
  - Replaced the speed Arduino Nano with an Arduino Nano BLE with built-in Bluetooth technology
  - Arduino Nano BLE made the sensors able to transfer data by Bluetooth
  - Design system to enable to provide power the sensors using 4 lithium AA batteries
  - MOSFET was added in the speed sensor circuit to save power
- Problems
  - Damaged Parts: Speed sensor controller had physical damage during implementation
    - Not able to send Bluetooth
    - Re-ordered another controller
- Experimentation and Validation
  - Measured the voltage and current used to play a full game
  - Measured the voltage and current used while the game is not being playing

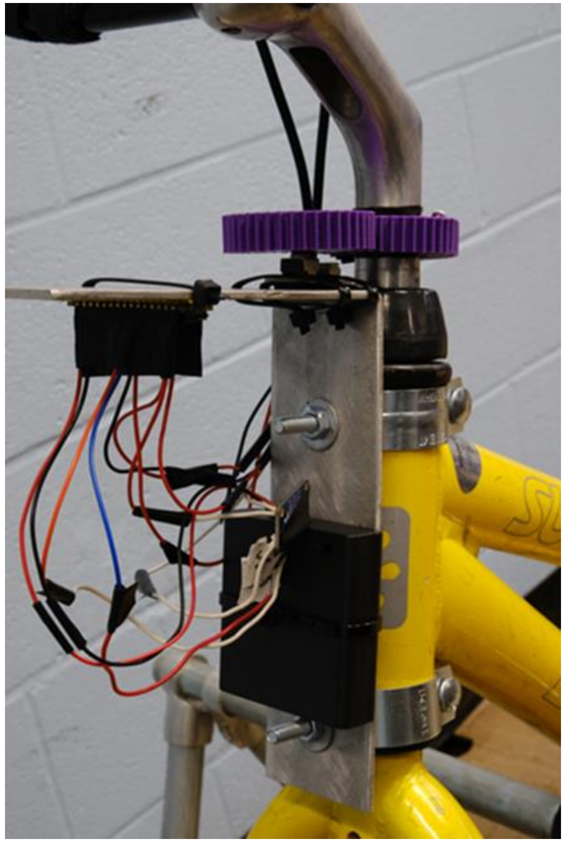
Bluetooth Low Energy (BLE) Communication

- Design and Implementation
  - Uses the ArduinoBLE library on the Nano, Bluepy library on the RPi
  - RPi initialized as Central device, Nano as Peripheral
  - Steering Service that holds Steering values, handlebar switches as characteristics
- Problems
  - Slight input delay
  - Occasional connection issues at launch
  - Sent as a byte, needed conversion

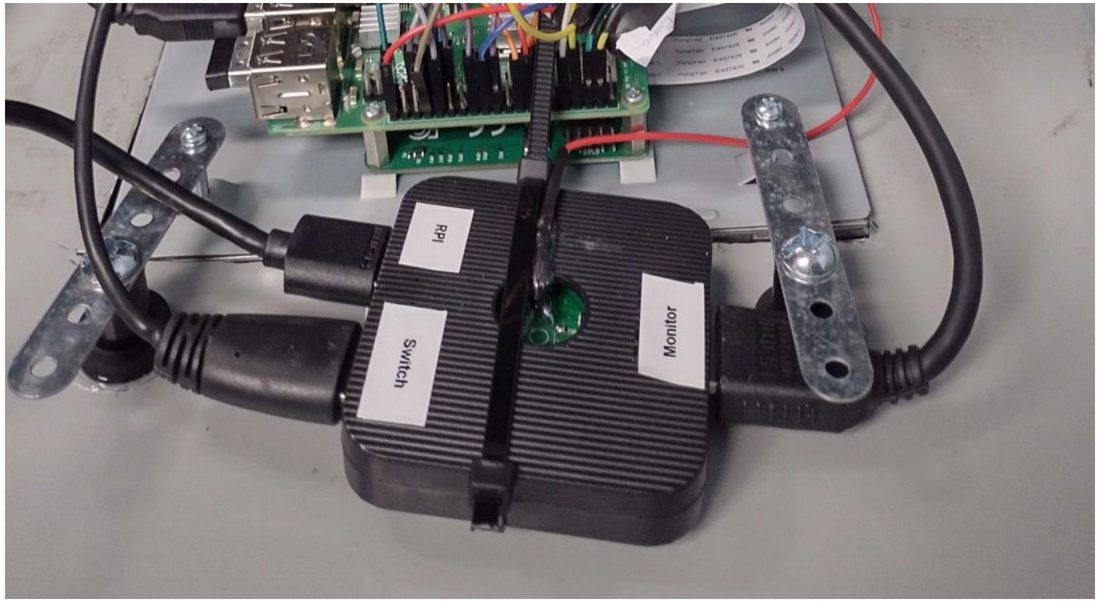
```
BLE.begin();
BLE.setAdvertisedService(steeringService);
steeringService.addCharacteristic(steeringCharacter);
steeringService.addCharacteristic(leftCharacter);
steeringService.addCharacteristic(rightCharacter);
BLE.addService(steeringService);
BLE.setLocalName("SteeringSensor");
BLE.setDeviceName("SteeringSensor");
BLE.advertise();
```

Steering Sensor Mount Enhancement

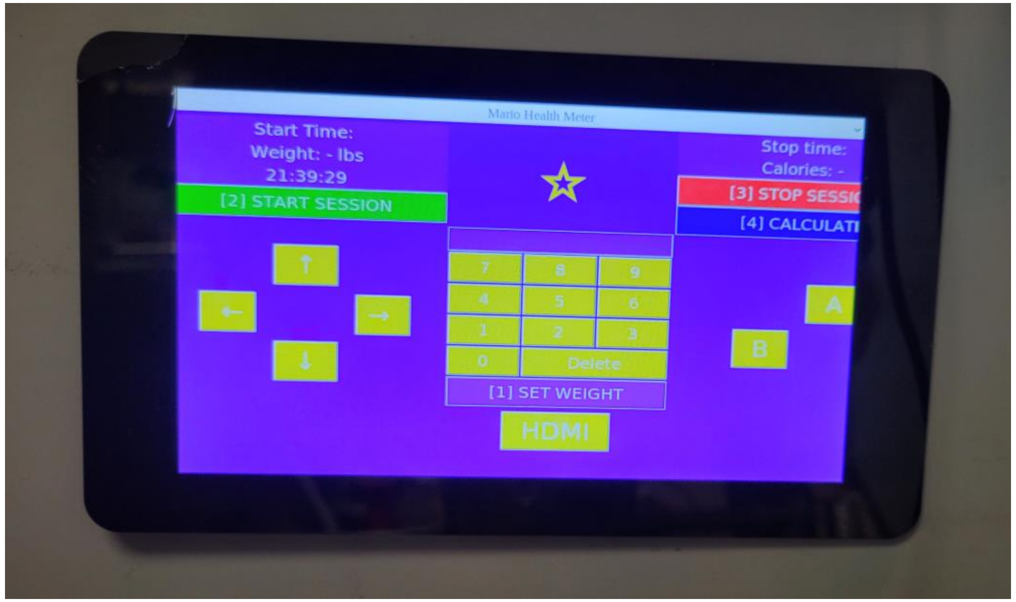
- Design and Implementation
  - Strengthened and stabilized steering sensor mount
  - Replaced existing plastic 3D printed L-Bracket with metal L-Bracket for enhanced strength
  - Attached L-Bracket to bike with conduit hanger for stability
- Problems
  - Order Delays: L-Bracket did not arrive on time
  - Increased Fabrication/Build Time: fabricated L-Bracket
  - Damaged Parts: Potentiometer was determined broken during testing
- Experimentation and Validation
  - Played multiple games to test the stability



HDMI Switcher



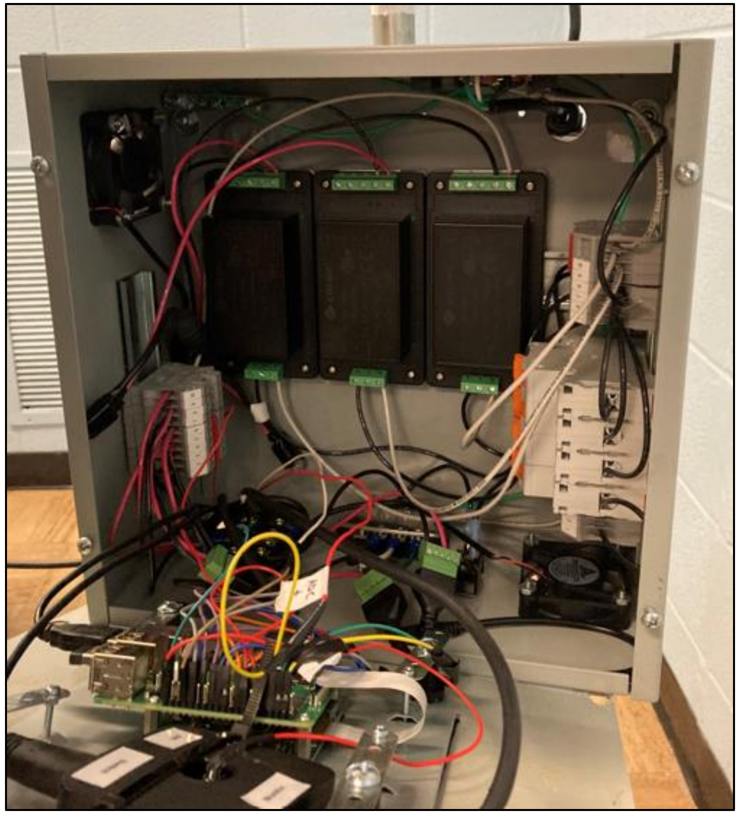
HDMI Switcher



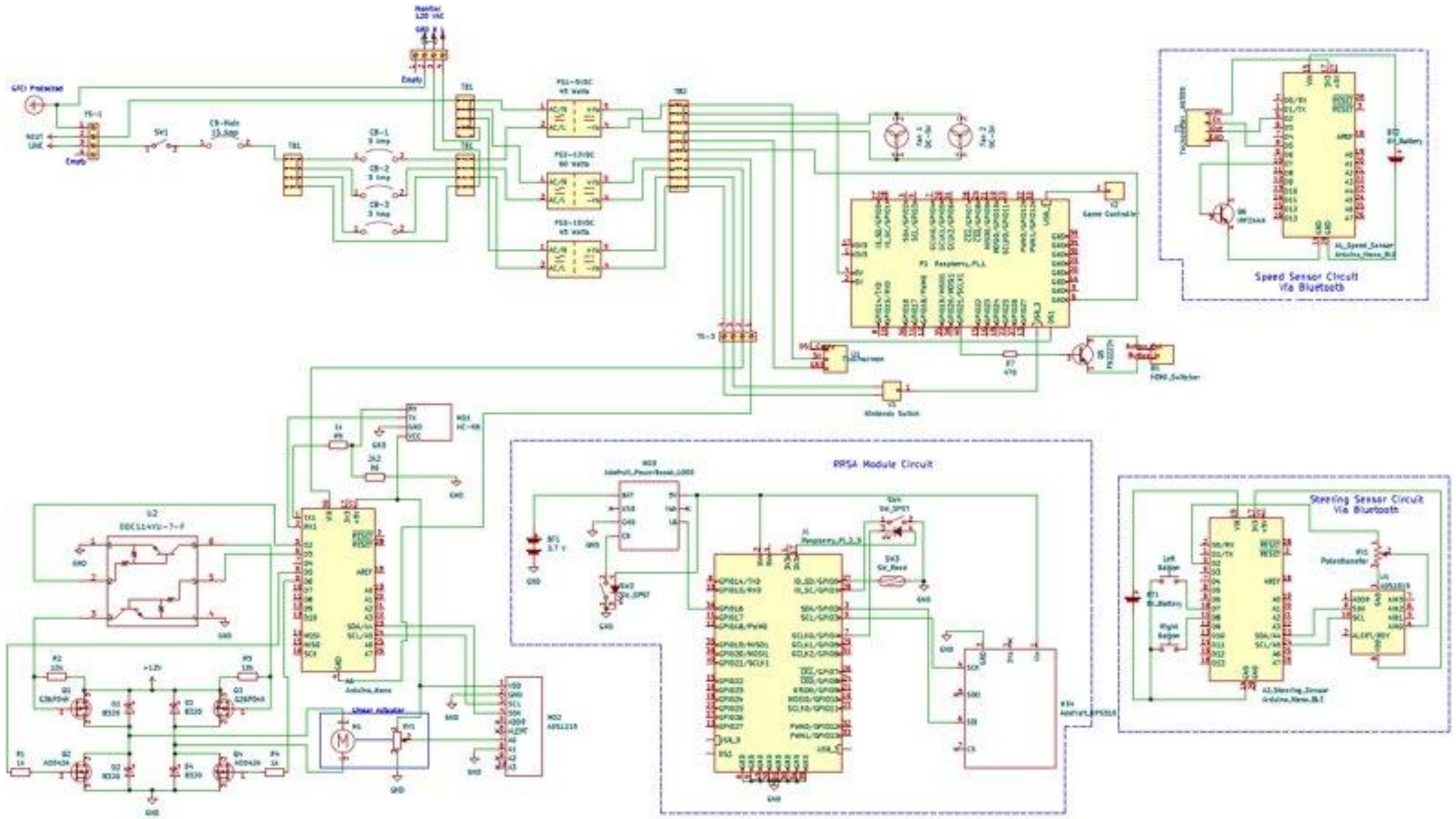
GUI w/
HDMI Switch

Power Distribution

- Design and Implementation
  - Provide device and system protection
  - Overprotection
  - Ground Fault Interruption (GFI)
  - Installation of over protection and GFI protection
  - Complete replacement of 120 VAC power supply
- National Electrical Code (NEC)
  - Overprotection devices to be at least 125% of continuous load
    - Article 210.20(A)
  - Wiring to be at least 125% of continuous load
    - Article 210.19(A)(1)



Final System Wiring Diagram



Budget

Subsystem	Cost
Resistance system	$ 188.02
Wireless Sensors	$ 142.98
Steering Bracket	$ 23.02
Power Distribution	$ 288.29
HDMI Switcher	$ 16.69
RRSA	$ 227.55
Linear Actuator Assembly	$ 254.09

Total Cost: $1,140.64

Future Work

- Enclose all Steering Sensor Subsystem equipment
- Replace tape on tire with paint or design new means of measuring speed
- Design means to prevent handlebars from turning 360 degrees
- Reduce flywheel size for further evaluation
- Design a means for adding magnets on both sides of the flywheel
- Find a means of decoding Nintendo switch Rumble data output
- Change frame rate of ride replay video based on speed

Questions?