Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

Team Piathon Report

Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

Table of Contents

Task Description **3**

Building the Robot (include photos) **4**

Parts / Sensors Used (include photo, and part numbers where applicable, such as HC-SR04 for the sonic echo sensor) **10**

How was the bot built (photos good to include) **11**

What libraries/software did you use in your code (include full reference) **14**

Pin Assignments **16**

Hardware Diagram **18**

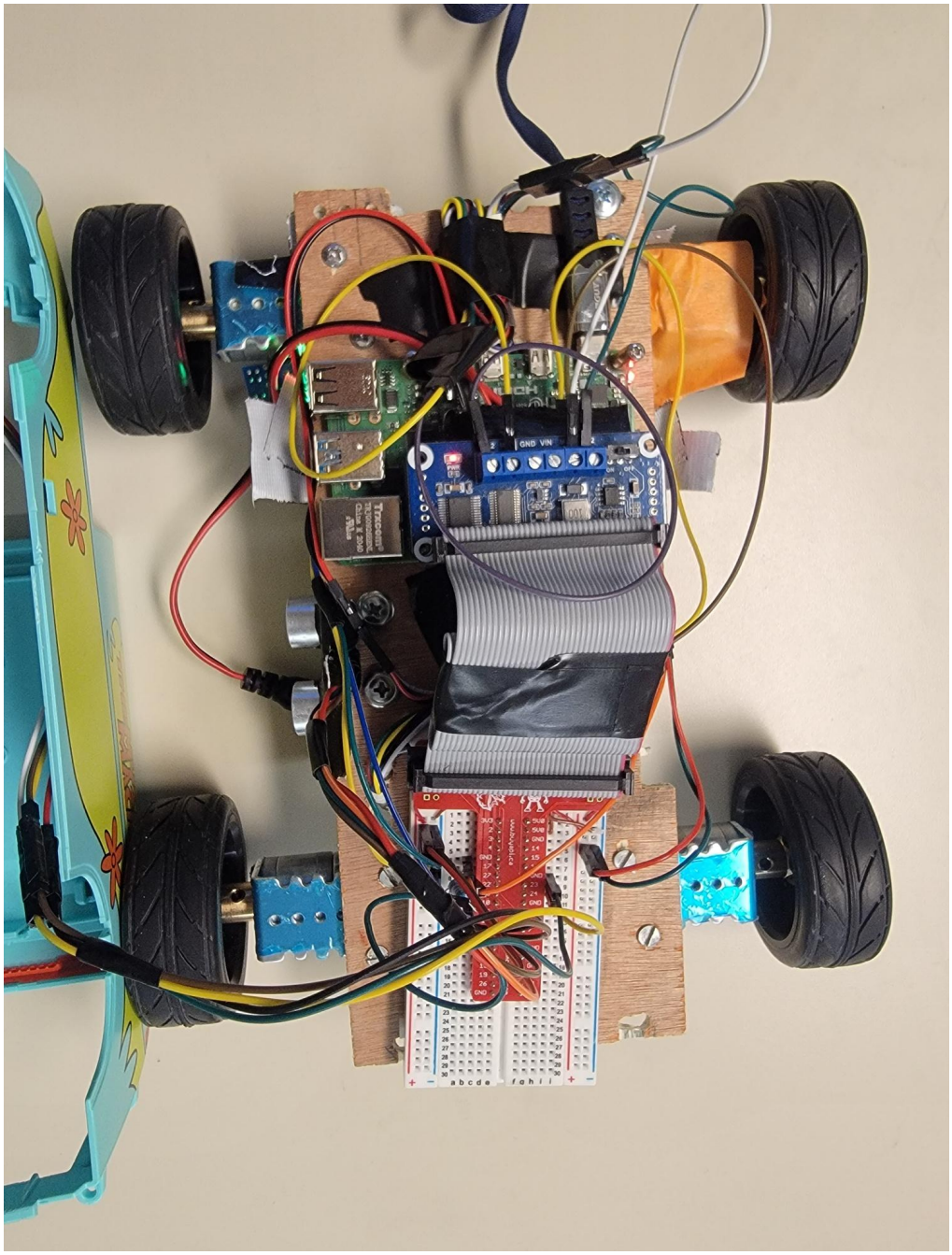
Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

Task Description

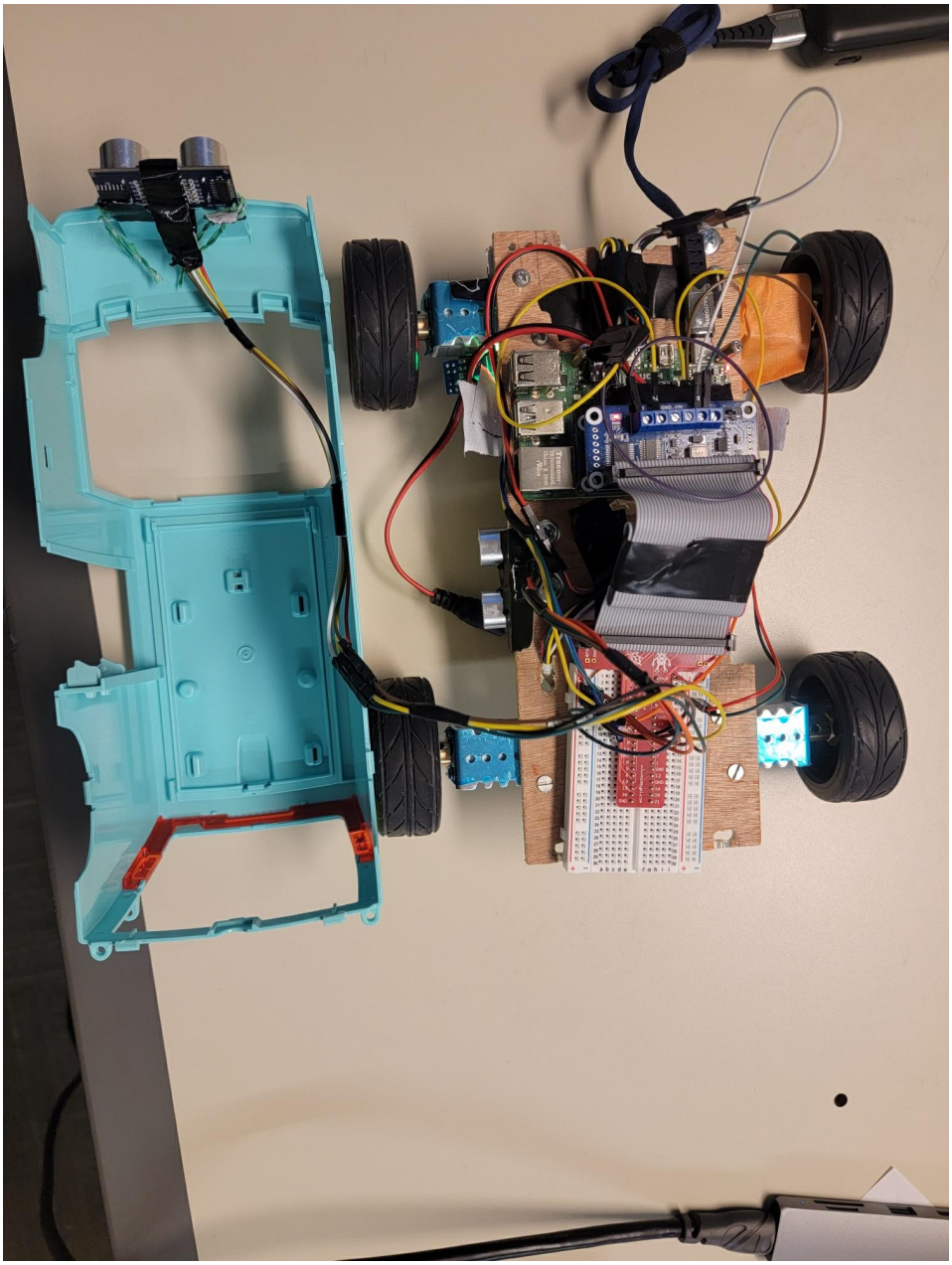
Create an autonomous vehicle that navigates itself using a black line that is on a white surface. The robot paths will have a variety of turns, examples include sharp turns and graduale turns in any direction. This Robotic Vehicle must also have the ability to detect objects obstructing its path and avoid an obstruction accordingly.

Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

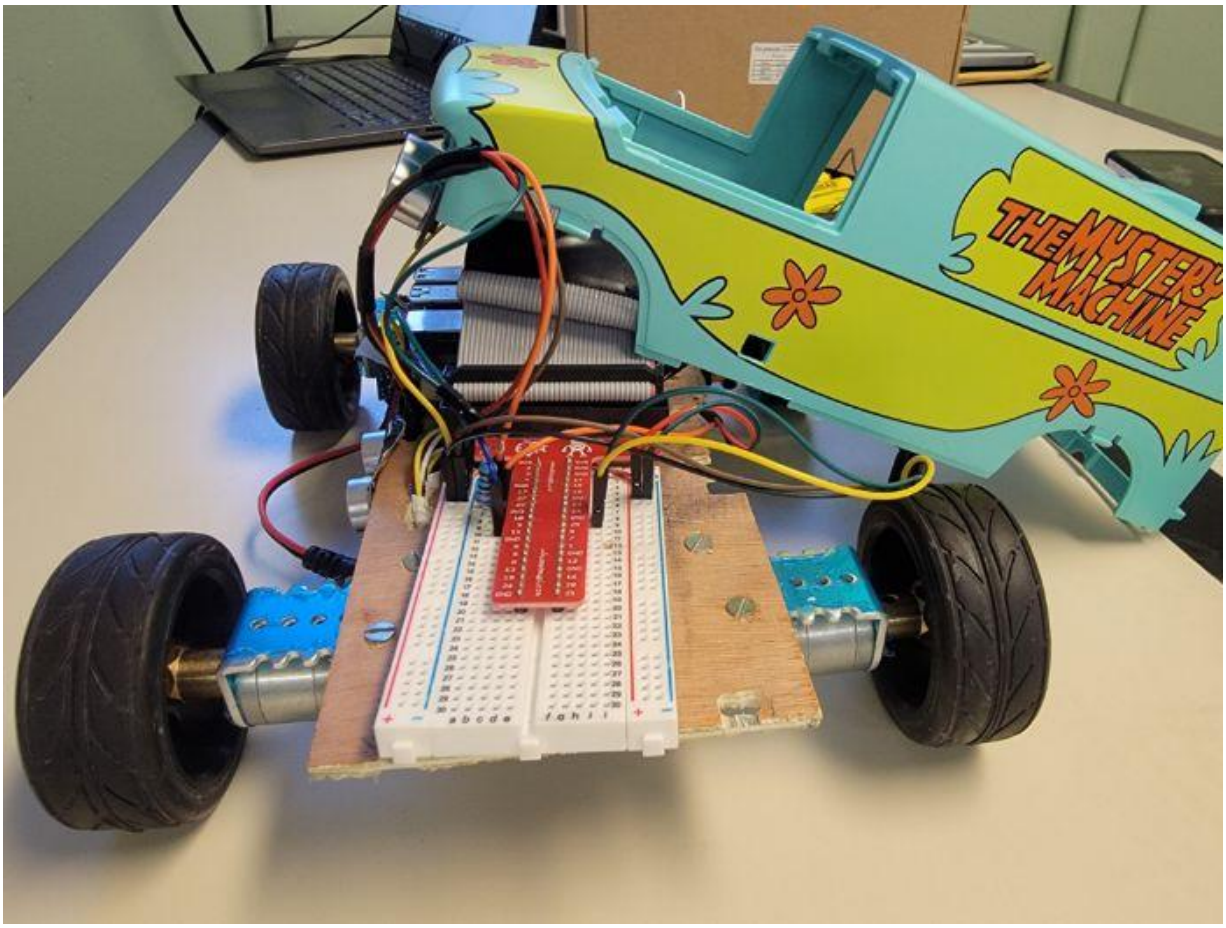
Building the Robot (include photos)



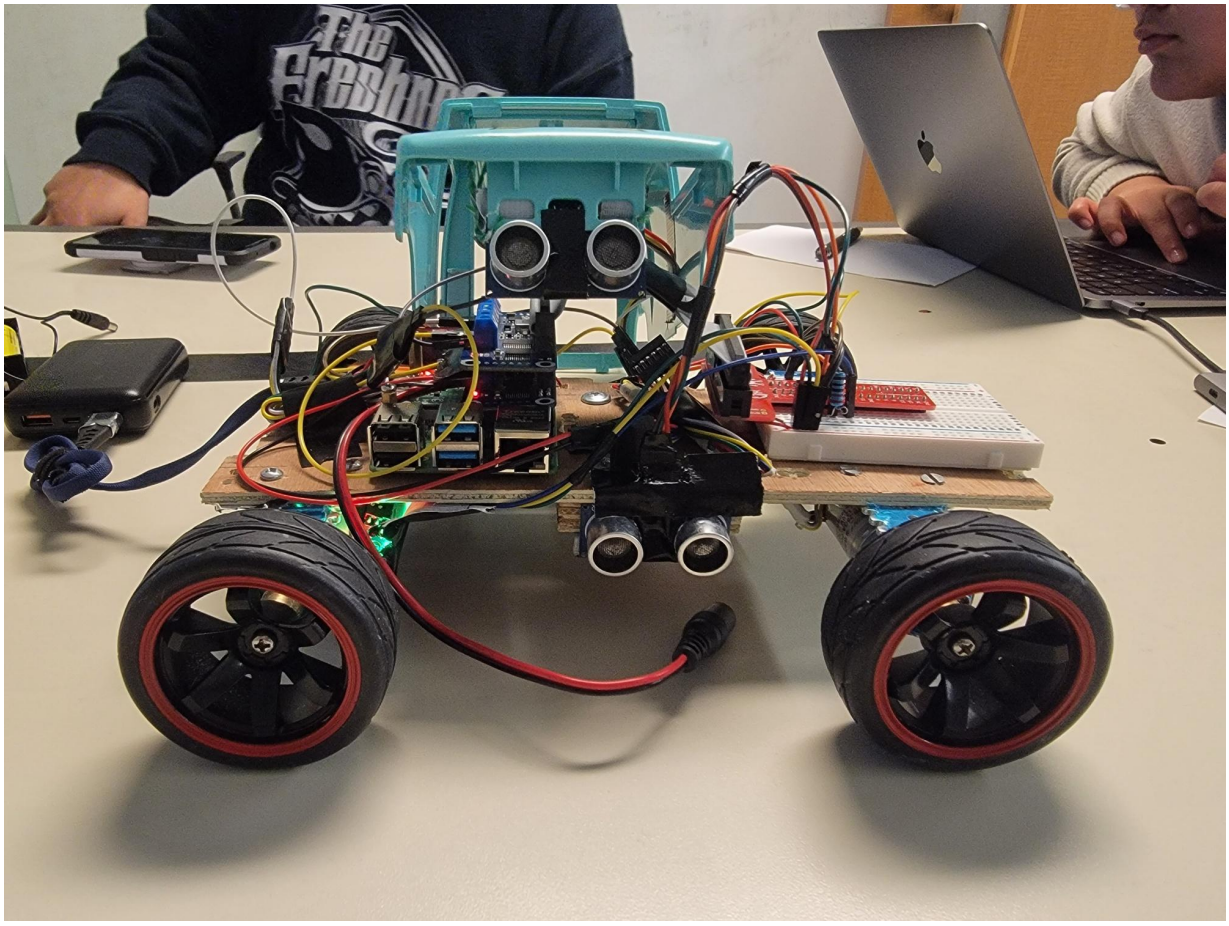
Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615



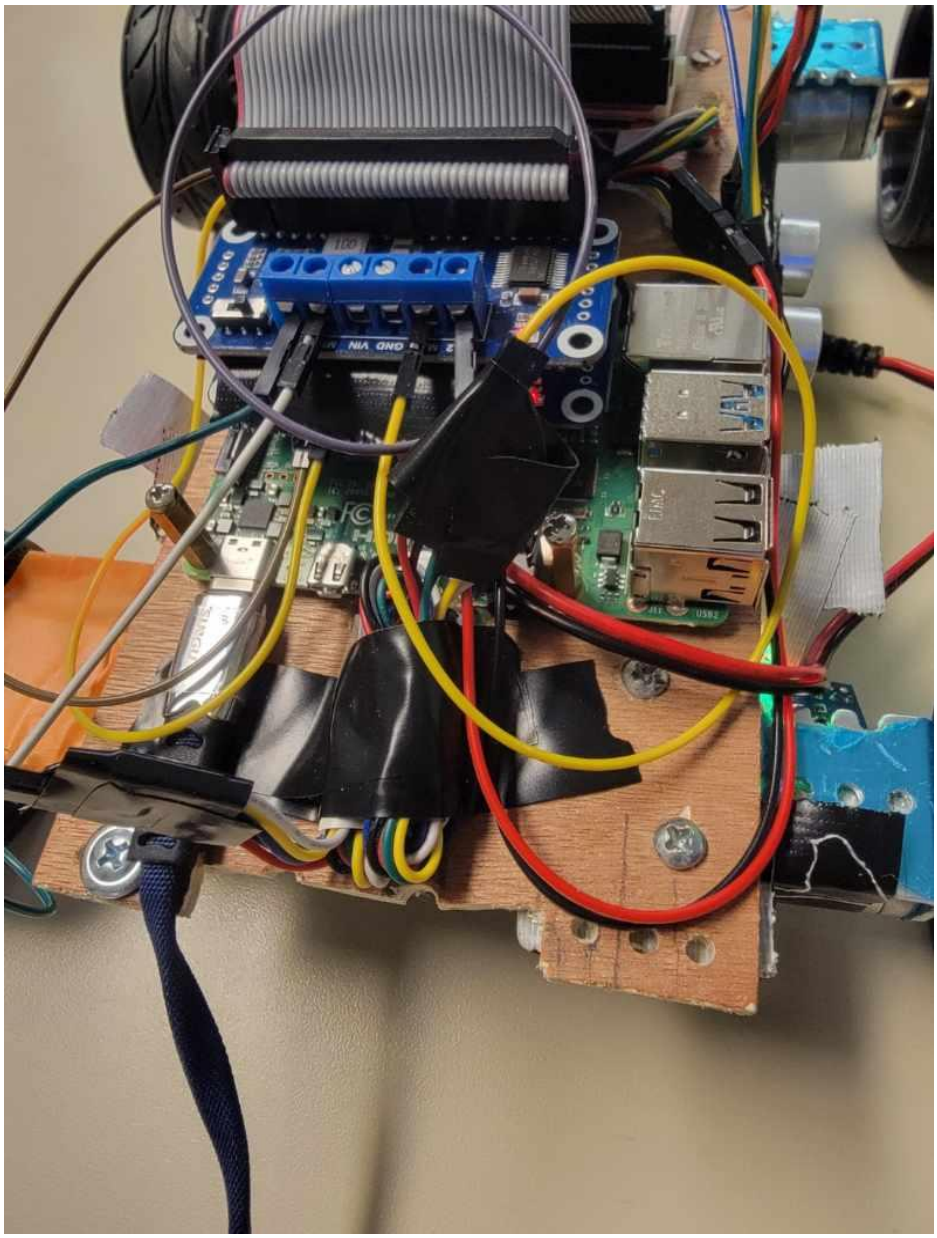
Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615



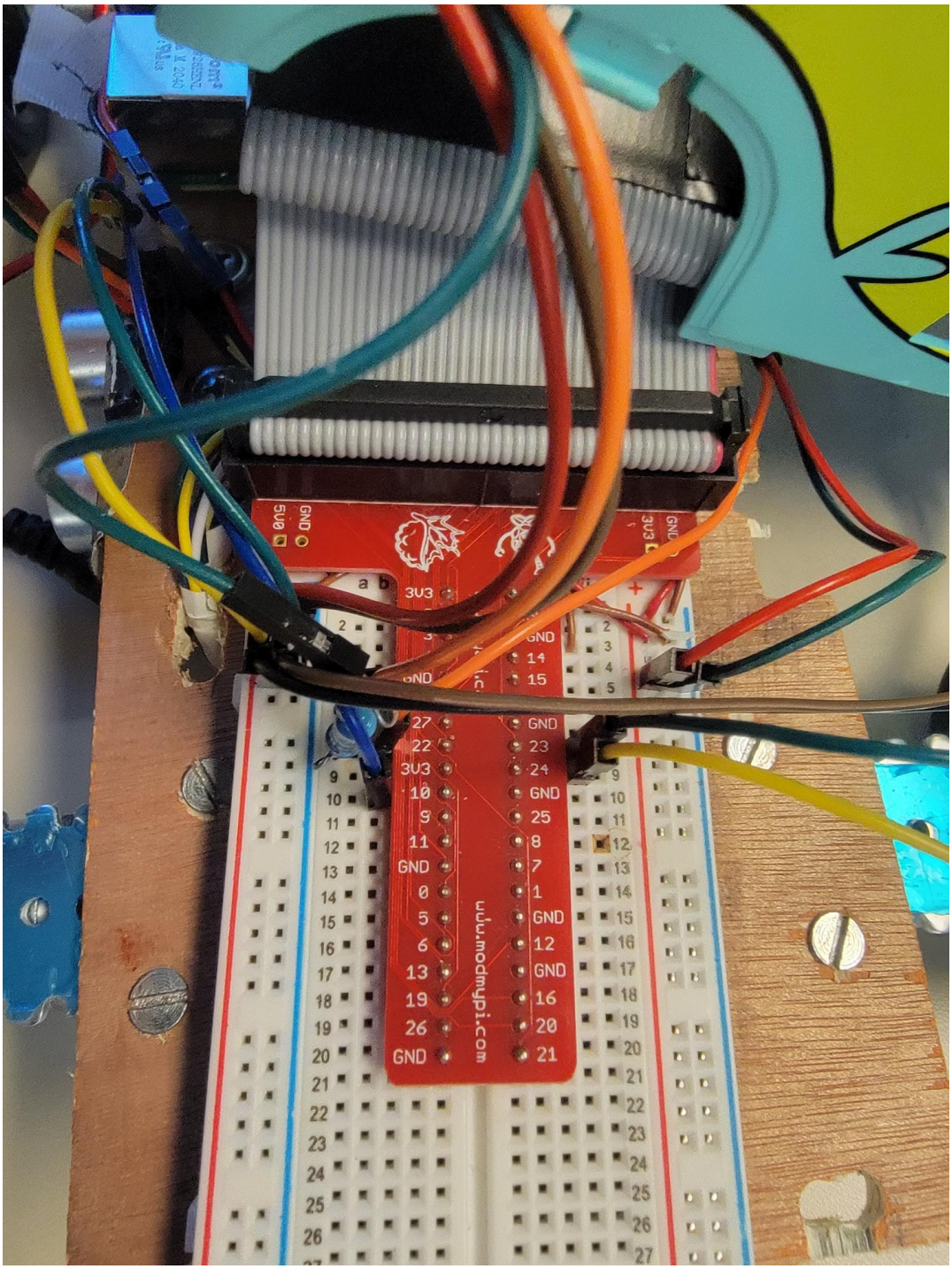
Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615



Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615



Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615



Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

Parts / Sensors Used (include photo, and part numbers where applicable, such as HC-SR04 for the sonic echo sensor)

Sensors:

- 2 HC-SR04 Echo sensor
    - Will be in front and side of our vehicle to determine distance of an object and avoid collision by telling the car to slow down.
- 2 Infrared Reflective Sensor
- sensor in front to determine if the car is either left or right of the line.
    - We will be placing one line sensor right, left and middle on the front of our vehicle to help us make sure the car stays on the line.

Motors:

- 4 Heavy Duty AWD (has Hall-effects sensor for speed)

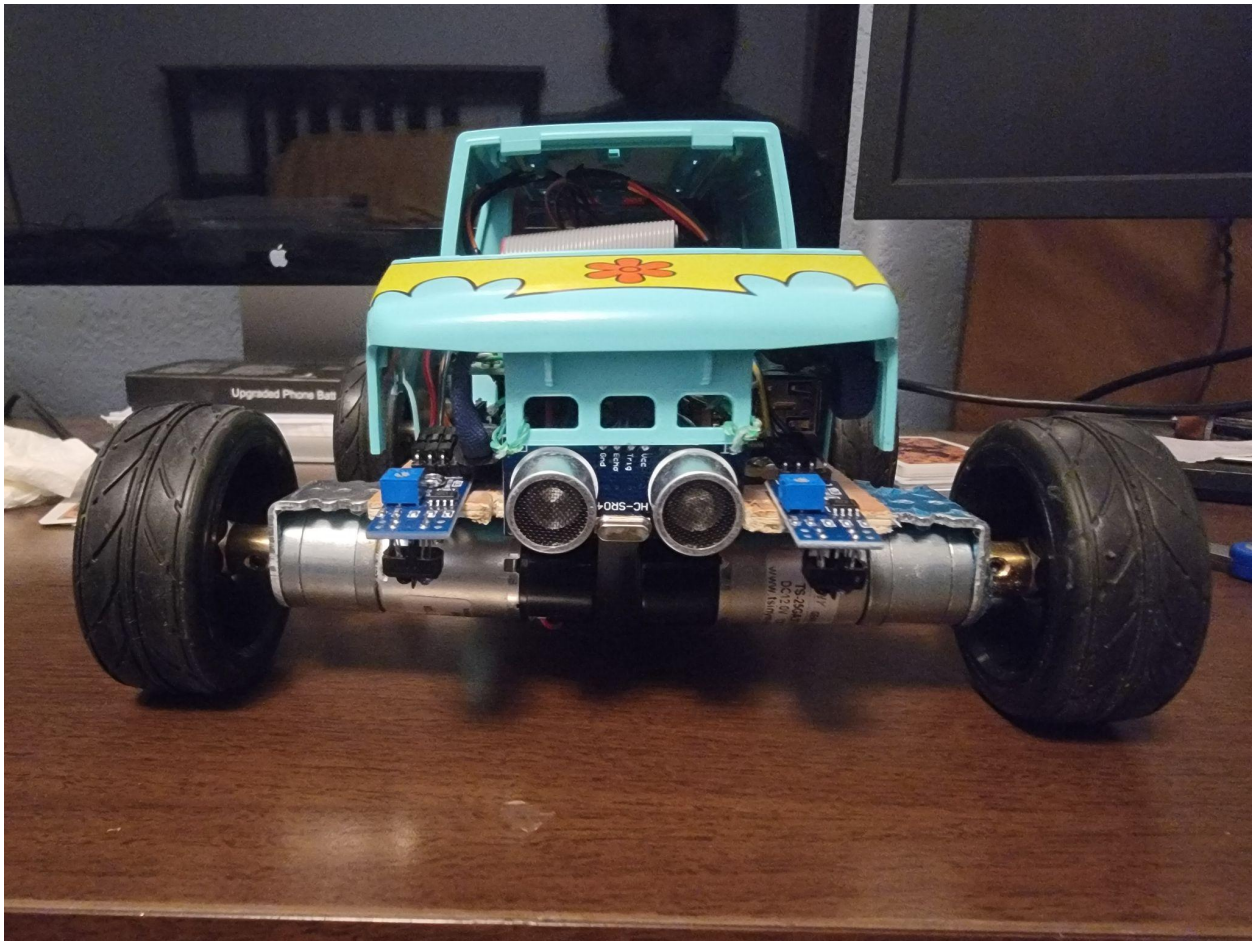
Components:

- Raspberry pi 4
    - The brain will keep everything in check.
- 2 Motor Shield
    - Will power the heavy duty motors and also powers up the pie.Two are stacked ontop of each other to give all 4 motors powers.
- Multiple resistors and wires
- Breadboard
- Ribbon Cable
    - Allows Motors and Pi to connect to the breadboard, and with the use of wires communicate to sensors and motors.
- Li-Ion battery

Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

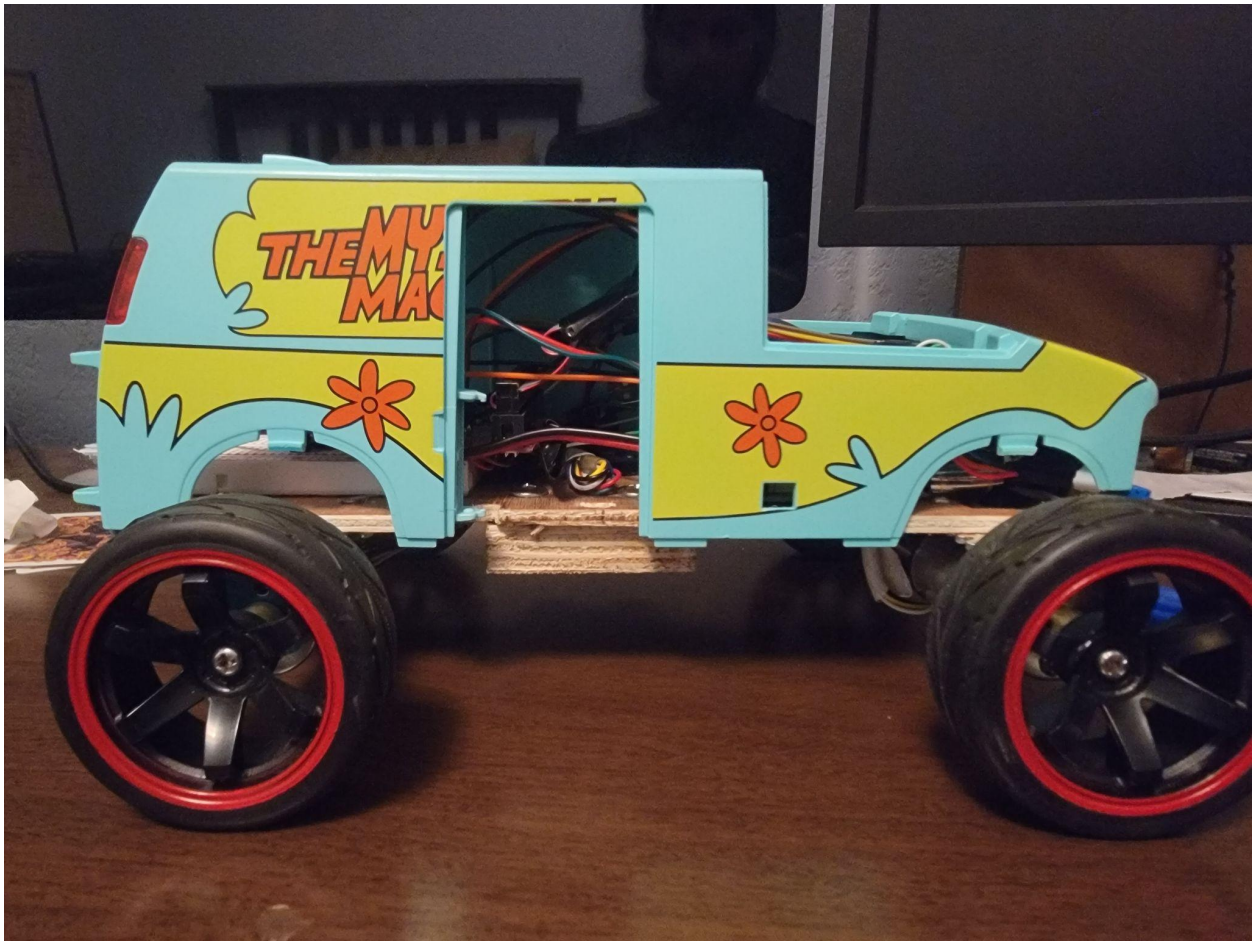
How was the bot built (photos good to include)

For our car we decided to have the Mystery Machine as our theme so we used a shell for the upper half of the car and we made the base out of wood where the components would sit on. We fixed four of the heavy duty motors to this wood base along with two echo sensors and two line sensors. One of the echo sensors was placed in the front of the car to help with stopping when it arrived in front of the object and the other sensor was placed on the left side of the car to help with going around the object. Both the left and right line sensors were placed under the car close to the front two motors and were being used to help us with following the black line on the track.



Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615



Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

What libraries/software did you use in your code (include full reference)

Flowchart of your code

1. For our line sensor code we had the car move forward when sensing white on both the left and right line sensor. We had the car turn left when the left line sensor saw the black line and the right line sensor saw white and turn right when the opposite conditions were true.
2. For our echo sensor code we would use our front echo sensor to stop a certain distance away from the object. After we made a complete stop we would use our front and left echo sensor to go around the object from the right side. We would use the front and left echo sensor to stay a certain distance away from the object until we have completed the object avoidance. After we finished the object avoidance we would restart the line sensor portion of our code and continue on as normal.

Libraries

- WiringPi
  - WiringPi
- Code from Waveshare for the motor hat
  - Motor Driver HAT - Waveshare Wiki

Software

- Windows/Mac OS
- Git/GitHub
- Atom

What worked well

- After getting over the hurdle of figuring out the logic behind the line sensors and when to move, stop, and turn. The performance of our car was consistent and handled a variety of turns well.

What were issues

- The motor hat we were using was causing confusion and miscommunication since it took up so many pins. Since it did that, there were overlaps with the Pi trying to use the sensors and motors, causing the logic to break.
- Following the lines around corners and turns required a lot of fine tuning and debugging with small
- Object avoidance caused a lot of work as we could not figure out the logic behind it quickly. It required multiple full days of work to debug and fine tune how we wanted the Echo sensors to find and move around the object.

Github Link: JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

Pin Assignments

MOTORS:

Serial Communication Bus:

PWM:

Set the speed for the Motors:

LEFTSIDE PCA_CHANNEL_0

RIGHTSIDE PCA_CHANNEL_5

Motor Hat 1:

Controlled front 2 motors

AIN1 PCA_CHANNEL_1

AIN2 PCA_CHANNEL_2

BIN1 PCA_CHANNEL_3

BIN2PCA_CHANNEL_4

Motor Hat 2:

Controlled rear 2 motors

AIN1 PCA_CHANNEL_1

AIN2 PCA_CHANNEL_2

BIN1 PCA_CHANNEL_3

BIN2PCA_CHANNEL_4

LINE SENSOR:

Left Line pins: 10

Right Line pins: 22

Echo Sensor:

Front Echo Sensor pins: Trig 23, Echo 24

Left Echo Sensor pins: Trig 17, Echo 27

[Github Link](): JGUZMAN95
Jocelyn Guzman: 915040482
Braden Dalit ID:917810002
Antonino Calabrese ID: 920831285
Anish Somisetty ID: 921956526
December 15th, 2021
CSC 615

Hardware Diagram