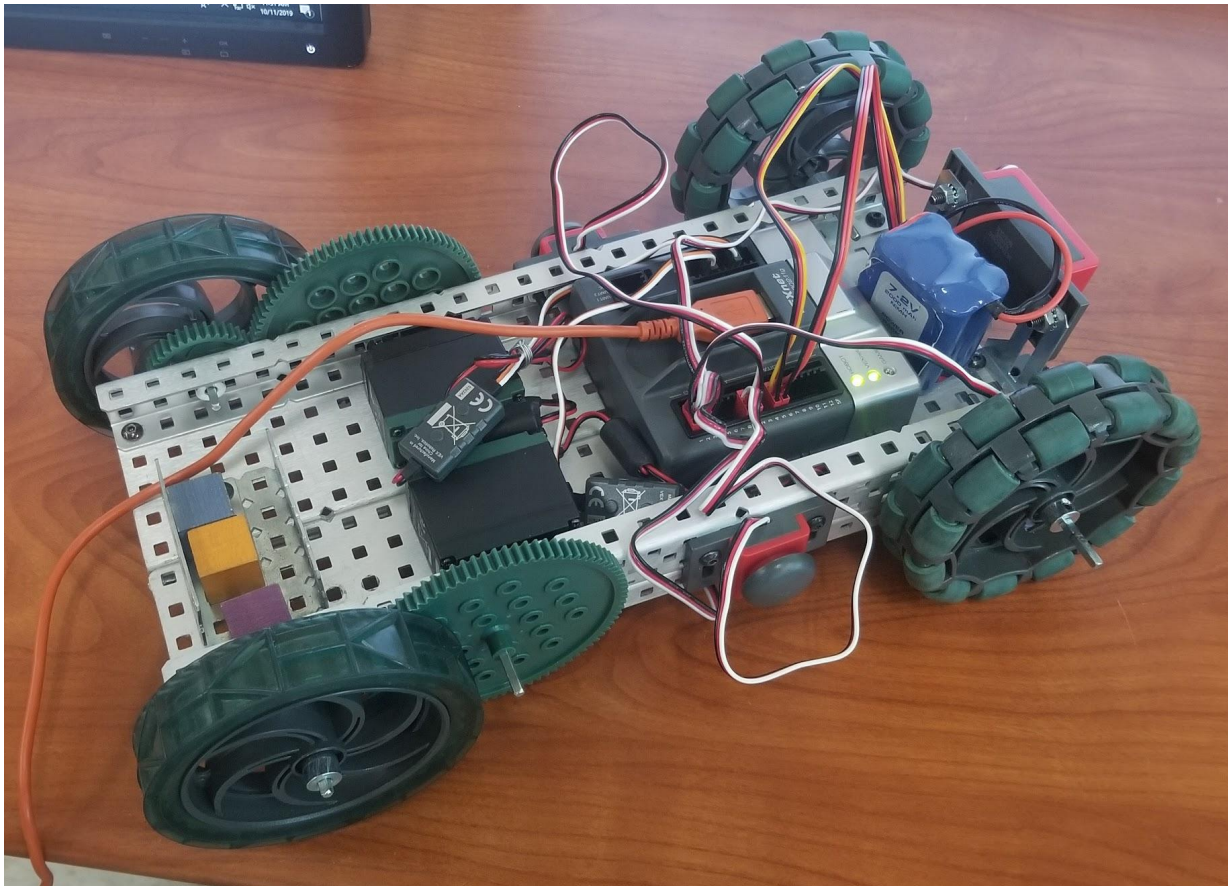
Activity 1.2.9 Automated Guided Vehicle

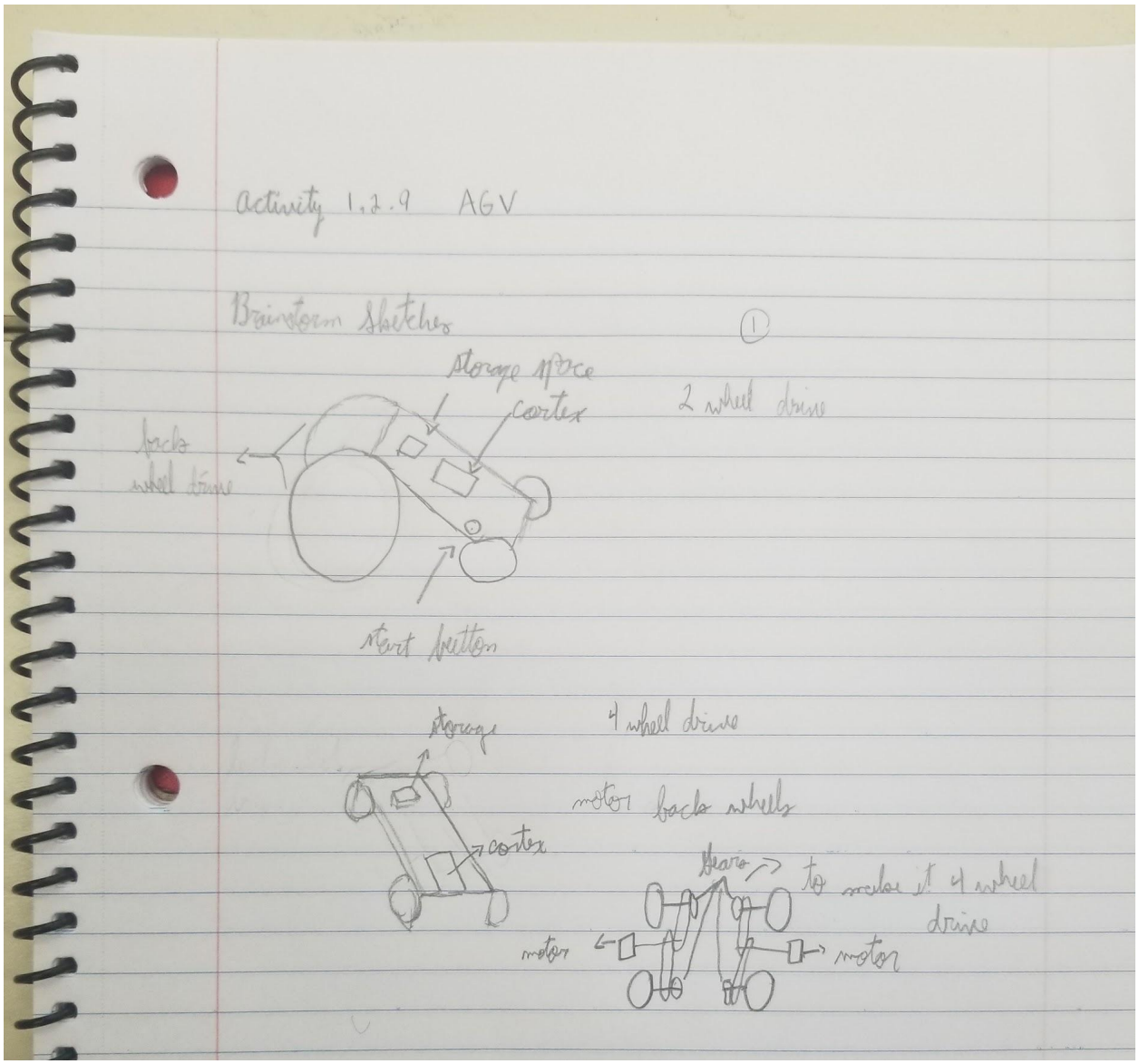


Jeremy Alayo, Ian Wilkinson, Handell Quiros, Norman Williams
Project Lead the Way | Computer Integrated Manufacturing
Mrs. Albarez
10-21-19

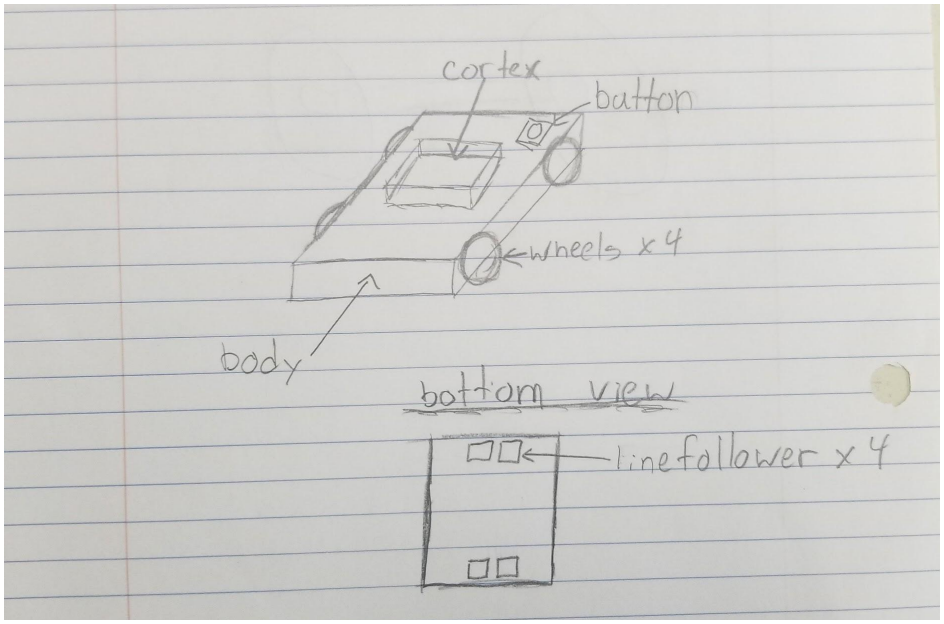
Table of Contents

Brainstorming Ideas/ Sketches

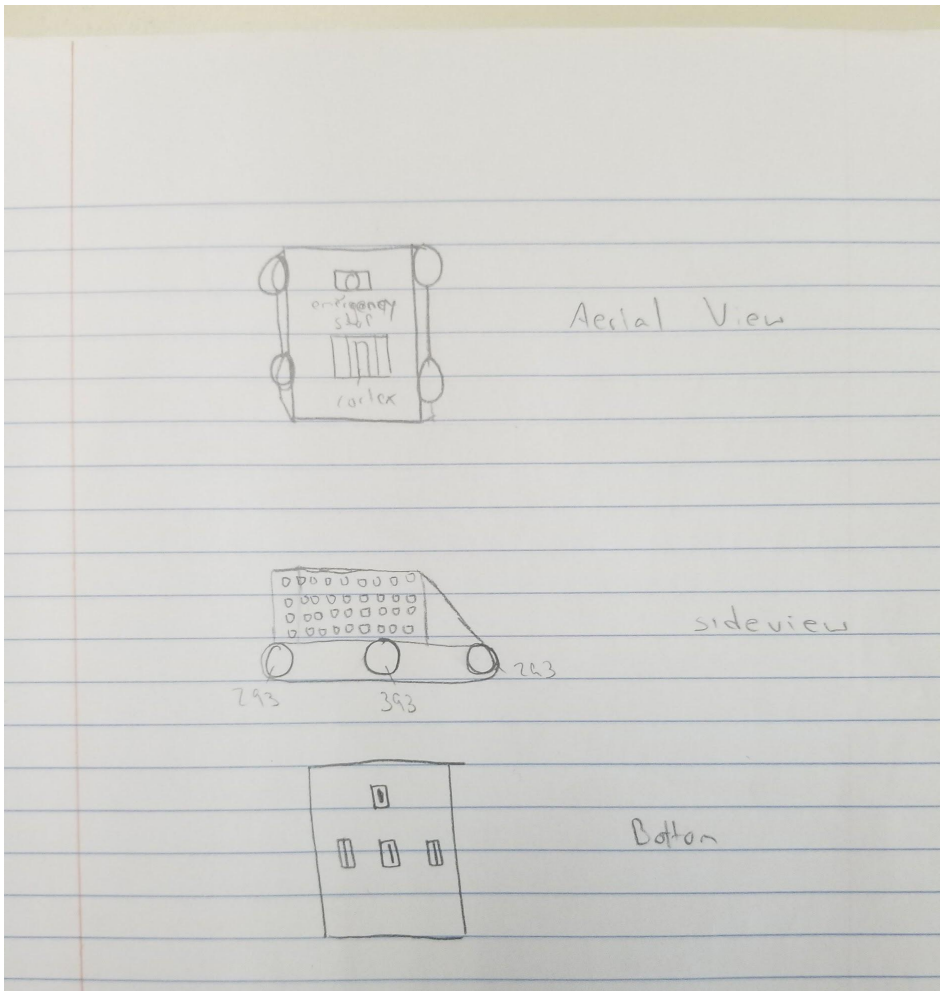
Design #1



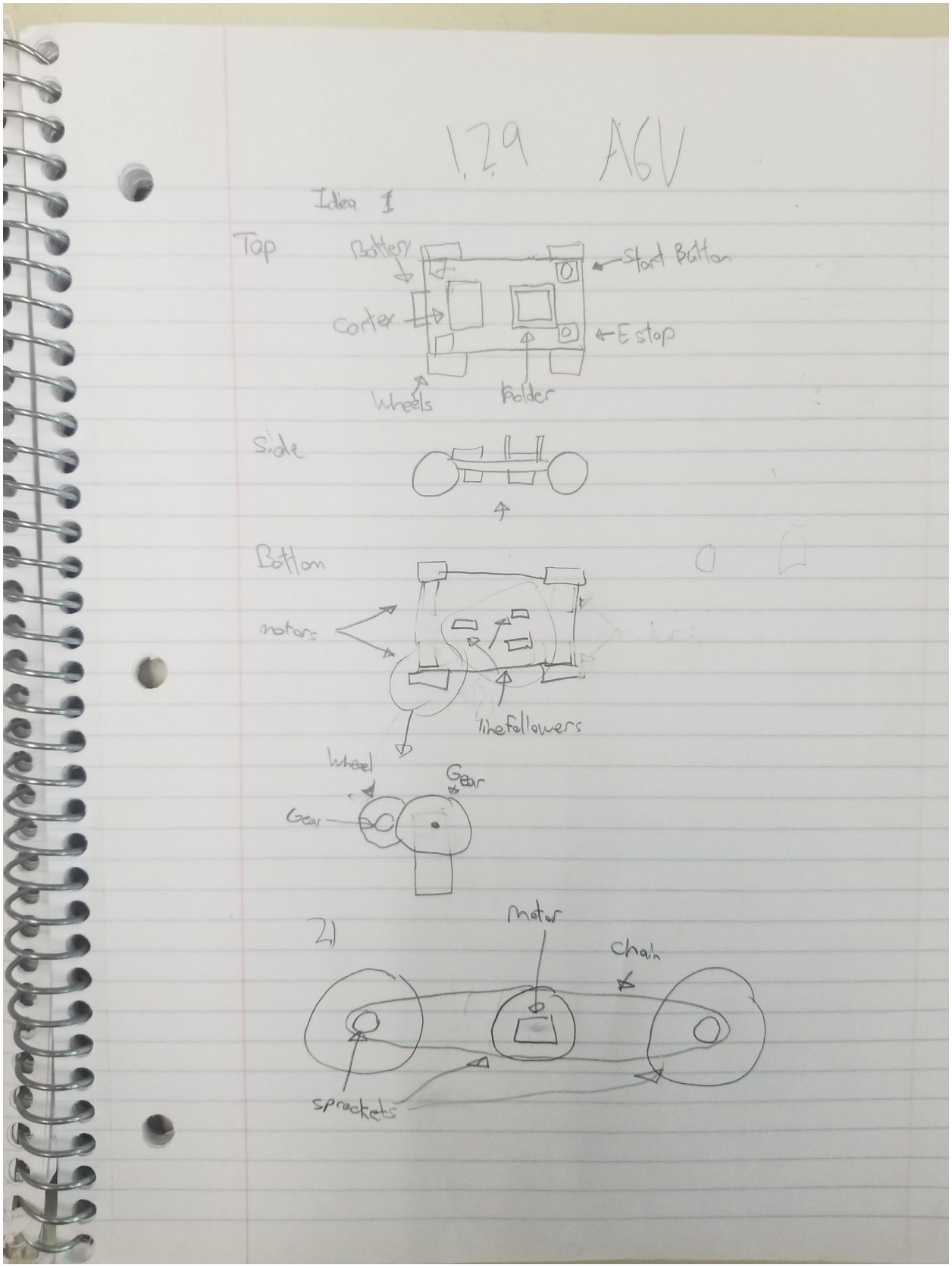
Design #2



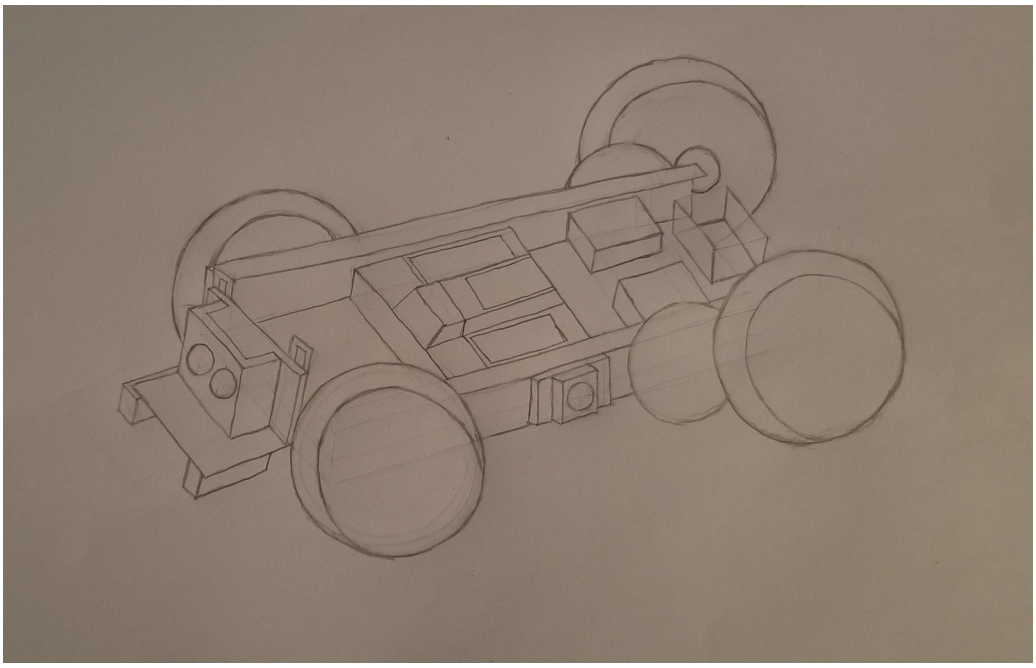
Design #3

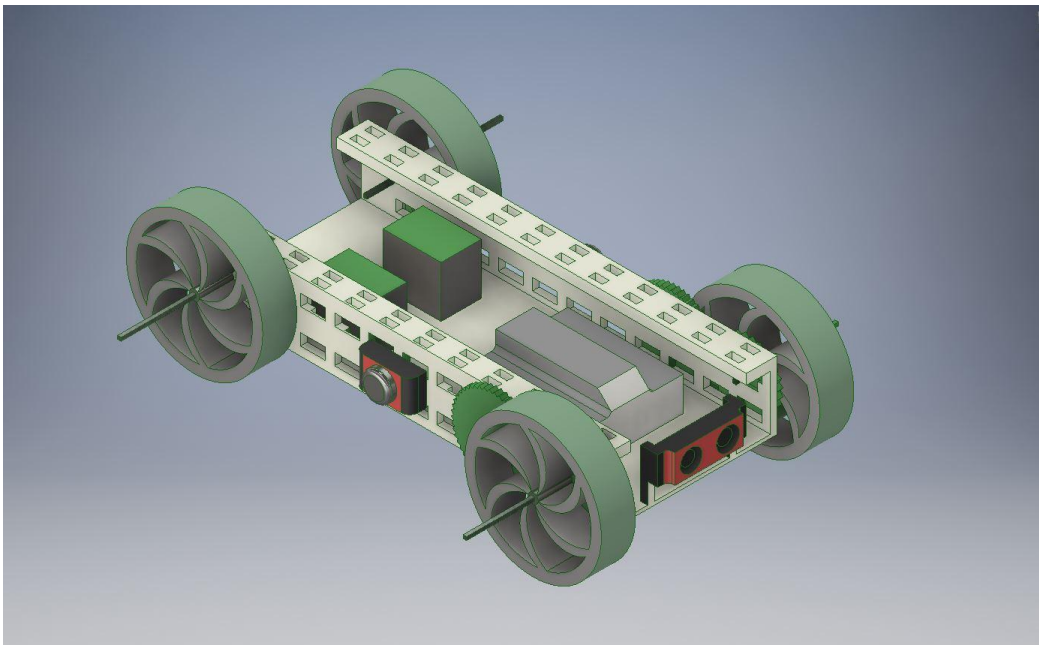


Design #4


1.29 AGV
Idea 1
Top
Battery
Start Button
Carter
E stop
Wheels
Holder
Side
Bottom
motors
linefollowers
Wheel
Gear
Gear
2.1
motor
chain
sprockets

Final Drawing





Code

```
task e_stop()
{
  while(1==1)
    {
      if(SensorValue(e_stopBtn)==1)
      {
       stopAllTasks();
      }
        wait1Msec(10);
  }
}
int bump;
task main()
{
  startTask(e_stop);
  {
    while(1==1)
    {
      if (SensorValue(startButton) == 1)
      {
      bump = 1;
      }
      while(bump==1)
        {
            if ((SensorValue(leftFollower) < 1500) && (SensorValue(rightFollower) < 1500))
            {
              startMotor(leftMotor,-40);
              startMotor(rightMotor,40);
            }
            if((SensorValue(rightFollower) > 1500) && (SensorValue(leftFollower) > 1500))
            {
              stopMotor(leftMotor);
              stopMotor(rightMotor);
              wait(5);
              startMotor(leftMotor,-23);
              startMotor(rightMotor,23);
              wait(0.12);
            }
            if(SensorValue(rightFollower) > 1500)
            {
              startMotor(leftMotor, -40);
              startMotor(rightMotor,-40);
            }
            if((SensorValue(leftFollower) > 1500) || (SensorValue(left2Follower) > 1500))
            {
              startMotor(leftMotor,52);
              startMotor(rightMotor,35);
              wait1Msec(1);
            }
            if(SensorValue(sonar) < 4)
            {
              stopMotor(leftMotor);
              stopMotor(rightMotor);
              wait(2);
            }
        }
    }
  }
}
```

Design Brief

Client: Mrs. Albarez

Target Consumer: Toys R Us

Designer: Ian Wilkinson, Handell Quiros, Norman Williams, and Jeremy Alayo

Problem Statement: Automated Guided Vehicles are efficient ways of transporting materials and products. They consist of at least one computer-controlled wheel-based load carriers and have a defined path that they can use to navigate.

Design Statement: We need to design, build, and program an Automated Guided Vehicle that follows a ¼" black line and carries cargo between stops using line followers for navigation.

Constraints:

- The AGV must have a start button and an emergency stop button.
- The AGV must not utilize more than three line followers.
- The AGV must stop at an intersection for six seconds.
- The AGV must detect an object that disrupts its path.

Parts List

Part Name	Quantity	Price
Shaft Collar (16 pack)	12	$7.99
Bumper Switch	2	$12.99
Ultrasonic Range Function	1	$29.99
Motor Controller 29 (4 pack)	2	$9.99
7.2V Robot Battery NiMH 3000mAh	1	$29.99
Line Tracker (2 pack)	3	$39.99
4” Omni-Directional Wheel (2pack)	2	$24.99
4” Wheel (4 pack)	2	$19.99
High Strength Shaft Bearing	4	$7.99
Plastic Spacer	6	$4.95
Nut 8 - 32 keps (100 pack)	22	$2.99
VEX ARM® Cortex® -based Microcontroller	1	$249.99
DriveshaftDrive shaft 12” (4 pack)	4	$8.96
2 wire motor 393	2	$ 17.99
Aluminum C-Channel 1x5x1x25 (6-pack)	2	$39.99
Aluminum C-Channel 1x2x1x25 (6-pack)	2	$29.99
Gear Kit (Includes 36 tooth & 28 tooth)	4	$12.99

Conclusion

In this activity, students were required to create an automated guided vehicle that would be able to follow a line. This AGV would be required to turn, to stop if an object was in its path, and to stop at intersections for 6 seconds and then proceed to move. We built our AGV with 4 wheels and 2 motors controlling the back 2 wheels. In the front, we put a sonar to be able to detect if there was an object in the way. One the bottom we first had 3 line followers positioned in a triangle with 2 sticking out of the front of the AGV on the left and right side and the other centered underneath the AGV. This was our first design, and our code was made so that the central line follower would be forced to always stay on the line. When approaching a turn the line followers sticking out would read if the line turned. The problem with this is that the AGV would not be good in slight turns. I decided to change the way we approached the problem. Instead of forcing the line follower to always stay on the line, the AGV would avoid the line with 2 line followers on the left and the right. The line follower would be placed so that the line was in between the line followers. The AGV would move forward until a sensor sensed something. If the line follower on the right detected the line the AGV would turn until it stopped sensing the line and the left side was the same. The AGV would stop if the line followers both sensed a line and if the sonar sensed an object close in front of it. Major problems that occurred were the code not being able to stay in the cortex, the code getting stuck in loops, the code taking to long to change from function to function, and the constant maintenance of the AGV build with screws becoming loose and wheels wiggling out.