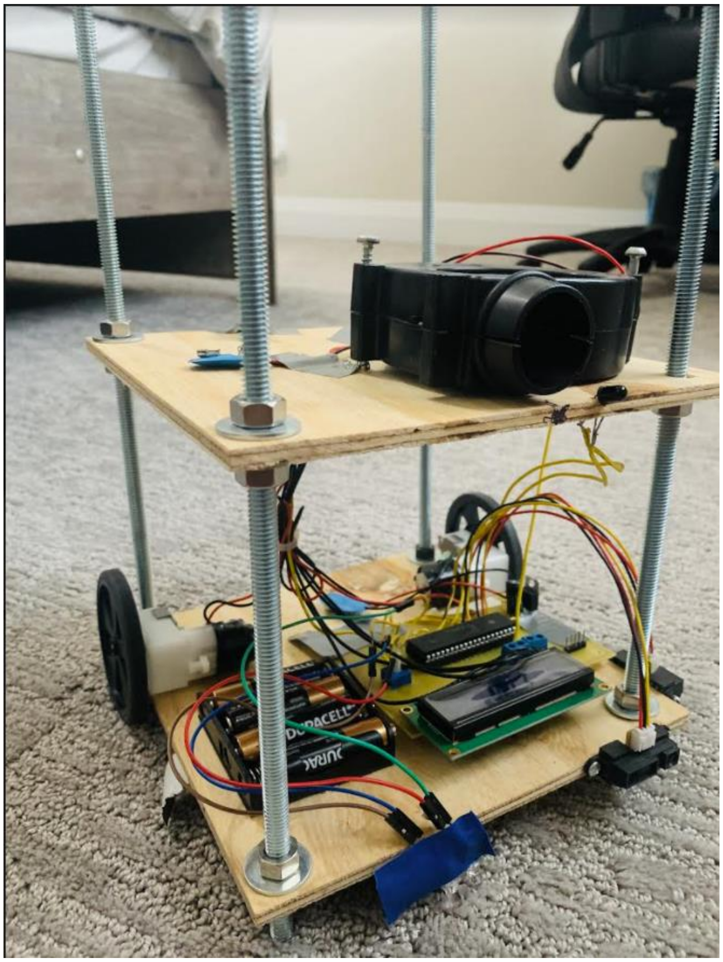
Firefighter Bot



By: Neel Patel
TEJ 4MI
Mr. Webb

Table of Contents

Intro to tech: 3

Grade 11 Experiences: 3

Grade 12 Experience: 4

Feedback on the course: 6

Advice: 6

Code: 8

Maze 1 Code: 8

Maze 2 Code: 12

Circuit Boards: 18

Photos: 20

Intro to tech:

I had very little experience in computer engineering before grade 11. I had not taken any technology courses in grade 9 or 10 because I believed I wanted to have a business-oriented career but in the middle of grade 10 my career aspirations changed, and I wanted to go into engineering. My parents had decided that it was a good idea to move to Waterloo for the rest of my high school years. My first orientation at SJAM was in the middle of my grade 10 year and the first classroom that I entered at SJAM was 2106 and I saw students working on one of their projects. They were soldering their circuit boards and at my previous school I had no idea that this type of course existed. I was so impressed with what these students were doing that I instantly knew that this is the type of career I wanted to go into. I spent the rest of my grade 10 year researching computer and mechatronics engineering. As soon as I moved to Waterloo, I searched for a local robotics team to join to get me more practical experience in addition to choosing computer engineering courses at SJAM.

Grade 11 Experiences:

My first tech course at SJAM was grade 11 computer engineering and can I just say it was a big shock how many things kids in the class already knew. I felt as if I was always a few steps behind the other kids when it came to tech class. I really did enjoy catching up though I felt like I was learning so many things in such a short amount of time. I had never soldered before the LED cube project and at the start it felt like I wouldn't even be able to get the soldering done. The more days that went by the farther behind I seemed to be falling and I wasn't really doing much better. But after a week passed, I felt like I was getting the hang of it and I started to catch up to other people in the class. The second part of this project was to build the circuit board on Traxmakr. I had trouble using Traxmakr at the start because I had never used such a program and there was quite a bit of a learning curve. However, by the end I felt pretty good with my skills on Traxmakr, and I had made my first circuit board. Unfortunately, the COVID-19 pandemic killed all the momentum I had built up in the past weeks and forced school to be online. I was very disappointed because not only, could I not finish the LED cube, but I wasn't even able to attempt the sumo bot. Even at home we were still given some work, but it was very minimal compared to what we could have done in class. The biggest assignment online was the LED cube program. I had some trouble at the start of writing the code because I was not taught how to program and had to learn entirely from home. It was also very difficult to picture what your program would do without having a LED cube to test it on. Visualization was an obstacle for me and to combat it I made a drawing of the LED cube to help me understand how to make certain patterns. This assignment was good practice for scenarios in which you won't have a physical object to program in front of you, but you still have to imagine what turning on certain ports will do for the project. This also allowed me to learn the programming language more thoroughly as I had to really understand the language to write a program for the cube without testing the program. Another technique I learned and used was pulse width modulation. Just reading what it did confused me a bit at the start but once I started to use it in my program it made more logical sense to me, and I understood why it would be so useful in many scenarios. In conclusion, grade 11 was a big jump for me and my understanding of computer engineering. Though it was disappointing that I could not finish my LED cube or attempt the sumo bot.

Grade 12 Experience:

I can say without a doubt that this course pushed me harder than any other course in the entirety of my high school years. There are many factors as to why this is the case for this course. One is my minimal experiences with these types of projects. When the Firefighter bot notes were put up, I had a read through of them and I felt so overwhelmed with the complexity of this bot. With no prior experience it made it difficult to imagine myself even finishing the bot and getting it into maze 1. We only had 11 days of in class time, so time management was crucial to getting the bot done. I experienced this firsthand as on the first and second days I did not feel I was getting as much done as I should have. The first day I started breadboarding I had no idea what I was doing because I had never used a breadboard in my life. It was a completely new experience, and this made it very difficult to even blink and LED. This got me thinking that if I wanted to make the most of my in-class time I was going to have to work at home. That night I prewired my breadboard for the motor components so I would not have to waste time in class. I thought this was a smart idea given the time restraints and it seemed to have paid off as I breezed through the motor component of breadboarding. The next component was tricky for me. It was the line detection, and it took me so long to do because I really didn't have a proper understanding of how it worked. For the first hour I thought it was a mistake I had made in either the breadboarding or the programming, but it seemed I didn't understand the phototransistor well and assumed that it was getting the light from the super bright LED, but it was not. The super bright had to almost be touching the phototransistor to activate it. This took the rest of my day to figure out and it felt like I was once again behind where I should have been. The next step was to attach the LCD to the breadboard. Again, to save time in class I had to prewire for the LCD at home and just come to school and attach it. I had also programmed for the LCD at home to save time. Once I got to school all I had to do was prepare the LCD by soldering on the pins and put it into my breadboard. I spent almost 20 minutes wondering why nothing was showing up on the screen even when my program was proven to be right. I soon realised that it was because my contrast was wrong, and the words blended in with the background. By slightly adjusting the contrast I could see the word on the screen. Over the coming weekend I programmed for wall detection, fire detection, and fire extinguish. Then I spent the rest of the in-class time diligently working through and completing all the breadboarding tasks. Unfortunately, I did not have the time to get my circuit boards printed in my in-class rime so I asked one of my friends in the other class to get me my printed circuit boards so I could work on them at home. I had downloaded PicBasic Pro and MPLAB X IPE onto my laptop so I would be able to program from home. This made working at home much easier. I also bought a soldering kit from Amazon that included a multimeter so I could solder and test components at home. Once I got my circuit boards, I wasted no time and I got to soldering on all the components. I did not rush this part as I wanted to make no mistakes that would lead me to having to replace components or the board as I was going to be at home for a couple weeks. I got all my components soldered onto my circuit board and I made sure I tested each one to make sure it was working. I completed most of the breadboarding tasks again but used my circuit boards to ensure everything would work. I then went to Lowe's to get wood and threaded rods to build my bot. Unfortunately, the only wood they had was pretty thin, so I had to make sure to use very small screws. The other unfortunate thing is I did not have a drill bit at home big enough to drill holes that would allow for the threaded rod to fit so I would have to wait till I was back in class to attach the two layers. That didn't stop me from building the layers separately though so I started off by trying to build a bottom layer that could do simple wall following and test it in my basement. This was not easy for me as I found out there was a problem with my PIC it seemed that it could only send 2 signals to the motors instead of 4 however, I found this out later so at the time I just switched the wires connecting to the terminal blocks so the two signals I sent out were for left forward and right forward. This meant that it could follow the and make left turns, but it would not be able to make right turns. For the second layer I just attached the fan, fan board, and infrared phototransistor. The display values were not correct, and I found out the reason for this was my windows as they were letting in the light from the sun, and this messed up the infrared phototransistors output. In a dark room it seemed to work fine. I could not test the fan because I did not have a 12-volt battery supply at home that I could use. So, I had to just hope it would work (It did not work). When I was back in school, I finally had the chance to attach the two layers and try my wall following. On the first run I was doing well until my bot ran straight in a wall because I forgot it couldn't do a right turn. I spent the rest of the day figuring out that there was a problem with my PIC and ended up having to replace it. As soon as I got that to work my LCD started acting up and it too had to be replaced. This was by far the hardest component to replace since there were so many pins soldered on. After that I got a good few runs in with just wall hugging and so I decided it was time to test the fan and wouldn't you know it didn't work. I discovered two main problems one was I put the wires from the fans incorrectly I did not know that it mattered but it did and second, I had a loose connection on the fan board. After I got these things figured out, I got the fan to start working. After that the next day I used my time to optimize my code by adjusting when to turn right and how close the flame must be to turn on the fan. Eventually I got to complete room 1 and room 2 near the end of the day. The next day I got to knockout room 3 and started trying for room 4. At first, I was having trouble with my line detection but as soon as I got it solved I had the smart idea to screw in my wheels but I used the wrong screw and broke my motor. I was sabotaging myself at this point and it took me over an hour to sort it all out. In the last 15 minutes of the day, I somehow completed room 4 with a lot of luck and it seemed that the day turned out alright. I had another couple weeks till the next time I would be in class and would attempt maze 2. This was very surprising to me since I never would have thought I would be done my bot let alone finished maze 1. At home I spent time strategizing on how to tackle room 2. I had taken pictures of the maze so I could visualize what programs would work. I did it on a room by room bases figuring out how to tackle each room individually. For room 1 my ideal strategy was to follow the line passing the first line and when it hit the second line the bot would reverse turn right and try to detect a flame if a flame was detected it would get closer to the flame otherwise it would turn left and move on to room 2. For room 2 the program would activate once it had to turn right after a certain amount of line counts it would move forward a certain distance then repeat the check flame maneuver of room 1. Room 3 was also coded somewhat like room two as it would enter into the room 3 program when it had a certain line count and had to turn right. To do this I had to add 1 to the line count if it did not detect a flame in room 2 thus adding an artificial line. The bot would proceed to turn right then move forward then turn left then go straight till its very close to the front wall. At this point it will turn left and check for a flame. This is as far as I got to programming and I did not have a clear plan for room 4. I thought I would wait to see how the other rooms go before programming for room 4. The first day back in class I only got to complete room 1 because my line detection was not reliable as the super bright LED and phototransistor moved around a lot. The next day I was still having problems because of my unreliable line detection so I had to use more tape to restrain the movement of the sensors.

Even then it was still not totally reliable but I got room 2. Room three was a bit more tricky because I had to adjust my program each run to make sure the bot moved the proper distance and the line detection was still shaky. By some chance of miracle I got room 3 on my last chance of the day and that made this entire course worth it. Even if I didn't get to room 4 I'm still proud of my accomplishments and for not believing I could get this bot done at the start, it was unbelievable to get this far.

Feedback on the course:

Let me start off by saying that this was the most challenging course in all of high school but as such it was also the most rewarding course. For me especially it felt like I got a lot of hands on experience and throughout the course I felt more confident in my abilities. It was such a different story at the start of the course where I felt completely lost and overwhelmed but now I feel as if I actually understand the bot and haven't just followed a step by step procedure to build it. The way this course is structured makes it so you have to set your own deadlines and find your own solutions to problems that occur. Being pushed to find and solve problems on my own helped me improve my troubleshooting techniques and gain more knowledge about the components. This hands-off approach to teaching forces students to expand their area of knowledge as they have to really understand how each component works. With very little steps to follow it becomes important that students create their own steps because without a clear plan it could be really hard to get this project done in time. All in all, this course being taught in a hands-off approach is the most beneficial to the student. Even Though it made the course tougher, I learned exponentially more than I would have if I had been given a step by step manual to building the bot.

Advice:

I have a couple tips for success in this course. The first tip is that at the start of the course make sure you read the notes. They may be very long but in the end you'll have to read them anyway because I can assure you that you will have done something wrong that you wouldn't have done if you read the notes. The second tip is if you're stuck on any part go read the notes again because even if you read them once it is unlikely you memorised them and reading them again will make this project much clearer. If you are still stuck after reading the notes then don't just wait around, ask the teacher how you can find what's wrong. Don't just ask them to check your circuit to find what's wrong because you won’t learn anything this way. Instead, you should ask the teacher to teach you how to troubleshoot problems. It is very important that you don;t wait around because each minute is important to getting this bot done. Just because you are ahead of the rest of your class doesn’t mean you should take a break. The moment you stop working you will fall behind. Another tip is to make the most of your time in class and out of class. In my class there was a very clear difference in progress between those students who worked at home and those who didn’t. Do as much as you can from home such as writing programs and making Traxmakr circuits. This will give you more class time to do things you can’t do at home. This project may seem hard at the start but as long as you put in the work you will find it to be very rewarding. In conclusion, make sure to work to the best of your abilities and you will definitely get a lot out of this course.

Code:

Maze 1 Code:

```
‘Start of by setting up all  LCD defines and variables
trisa =%11111111                        ‘Set port a to input
trisb =%00000000                        ‘Set port b to output
trisc =%11111111                        ‘Set port c to input
trisd =%00000000                        ‘Set port d to output
DEFINE LCD_DREG PORTD            ‘LCD setup
define LCD_DBIT 4
define LCD_RSREG PORTD
define LCD_RSBIT 0
define LCD_EREG PORTD
define LCD_EBIT 2
define LCD_RWREG PORTD
define LCD_RWBIT 1
define LCD_BITS 4
define LCD_LINES 2
define LCD COMMANDUS 1500
define LCD_DATAUS 44
define ADC_BITS 8
define ADC_CLOCK 3
define ADC_SAMPLEUS 50
frontwall var word                      ‘Front wall in cm
leftwall var word                       ‘Left wall in cm
flamedistance var word                  ‘Flame distance in cm
linecount var word                      ‘Line count
linecount = 0                           ‘Set Line count to 0
x var byte                              ‘Front wall initial input
y var byte                              ‘Left wall initial input
z var byte                              ‘Flame distance initial input
ADCON1 = 0

main:
adcin 0, x                              ‘Assign sensor values to variables
adcin 1, y
adcin 2, z
frontwall = (((6787/(x-3)))-4)/5        ‘Convert raw values to cm
leftwall = (((6787/(y-3)))-4)/5
flamedistance = z
LCDout $FE, 1, #linecount               ‘Display line count
if z <= 5 then goto flame               ‘Check for flame
if portc.0==0 then goto line                    'Check of line

while leftwall <18 and leftwall >12 and frontwall > 18        'Go Straight while conditions are
met
   if portc.0==0 then goto line                               'Check for line

   portb =%00000110                                           'Tell motors to go straight
   pause 200
   adcin 0, x                                                 'Update variables
   adcin 1, y
   adcin 2, z
   frontwall = (((6787/(x-3)))-4)/5
   leftwall = (((6787/(y-3)))-4)/5
   flamedistance = z
   if z <= 5 then goto flame                                  'Check for flame
WEND

adcin 0, x                                                    'Update variables
adcin 1, y
frontwall = (((6787/(x-3)))-4)/5
leftwall = (((6787/(y-3)))-4)/5

while leftwall < 12 and frontwall > 18                              'Tilt Right while conditions
are met
   if portc.0==0 then goto line                               'Check for line

   portb =%00000100                                           'Tell motors to tilt right
   pause 200
   adcin 0, x                                                 'Update Variable
   adcin 1, y
   adcin 2, z
   frontwall = (((6787/(x-3)))-4)/5
   leftwall = (((6787/(y-3)))-4)/5
   flamedistance = z
   if z <= 5 then goto flame                                  'Check for flame
WEND


while leftwall > 18 and frontwall > 18                              'Tilt Left while conditions are
met
   if portc.0==0 then goto line                               'Check for line

   portb =%00000010                                           ' Tell motors to tilt left
   pause 200
```

```
  adcin 0, x                                          ‘Update variables
  adcin 1, y
  adcin 2, z
  frontwall = (((6787/(x-3)))-4)/5
  leftwall = (((6787/(y-3)))-4)/5
  flamedistance = z
  if z <= 5 then goto flame                           ‘Check for flame
WEND

while frontwall <= 15                                 ‘Turn Right while conditions are met
  if PORTC.0==0 then goto line                        ‘Check for line
  portb =%00000101                                    ‘Tell motors to turn right
  pause 200
  adcin 0, x                                          ‘ Update variables
  adcin 1, y
  adcin 2, z
  frontwall = (((6787/(x-3)))-4)/5
  leftwall = (((6787/(y-3)))-4)/5
  flamedistance = z
  if z <= 5 then goto flame                           ‘Check for flame
  if linecount == 6 then goto room4                          ‘Check if bot needs to go to
room 4
WEND
if portc.0==0 then goto line                          ‘Check for line

goto Main                                             ‘Go back to start of main

flame:                                                ‘Flame extinguish
portb =%00010101                                      ‘Turn on fan while spinning
pause 200
portb =%00010000                                      ‘Keep fan on but motors off
       pause 800
goto flame                                            ‘Endless loop

line:                                                 ‘Line count update
linecount =  linecount+1                              ‘Add one to line count
LCDout $FE, 1, #linecount                             ‘Display line count on LCD
pause 400
goto main                                             ‘Go back to main

room4:                                                ‘Room 4 program
portb =%00000101                                      ‘Turn 180 degrees
pause 750
portb =%00000110                                      ‘Go straight for a bit
```

```
pause 1000
goto main                                          ‘Go back to main to fin flame
```

Maze 2 Code:

```
‘Start of by setting up all  LCD defines and variables
trisa =%11111111                                            ‘Set port a to input
trisb =%00000000                                            ‘Set port b to output
trisc =%11111111                                            ‘Set port c to input
trisd =%00000000                                            ‘Set port d to output
DEFINE LCD_DREG PORTD                                  ‘LCD setup
define LCD_DBIT 4
define LCD_RSREG PORTD
define LCD_RSBIT 0
define LCD_EREG PORTD
define LCD_EBIT 2
define LCD_RWREG PORTD
define LCD_RWBIT 1
define LCD_BITS 4
define LCD_LINES 2
define LCD COMMANDUS 1500
define LCD_DATAUS 44
define ADC_BITS 8
define ADC_CLOCK 3
define ADC_SAMPLEUS 50
frontwall var word                                          ‘Front wall in cm
leftwall var word                                           ‘Left wall in cm
flamedistance var word                                      ‘Flame distance in cm
linecount var word                                          ‘Line count
linecount = 0                                               ‘Set Line count to 0
x var byte                                                  ‘Front wall initial input
y var byte                                                  ‘Left wall initial input
z var byte                                                  ‘Flame distance initial input
ADCON1 = 0

main:
if linecount = 0 then portb =%00000000                      ‘Stop all B port outputs at the start
adcin 0, x                                                  ‘Assign sensor values
adcin 1, y
adcin 2, z
frontwall = (((6787/(x-3)))-4)/5                            ‘Convert wall sensor values to cm
leftwall = (((6787/(y-3)))-4)/5
flamedistance = z                                           ‘Assign value to flamedistance
if linecount = 0 then goto turnleft                         ‘Start attempt with left turn
if linecount == 3 then goto room1                           ‘When line count is 3 enter room 1

LCDout $FE, 1, #linecount                                   ‘Show linecount on LCD
```

```
if z <= 5 then goto flame                                   'If flame is within 5 cm start fan
if portc.0==0 then goto line                                'Line count

while leftwall <18 and leftwall >12 and frontwall > 18      'Go Straight while conditions are
met
   if portc.0==0 then goto line                             'Check for line
   if linecount == 3 then goto room1                        'Check if bot needs to got to room 1

   portb =%00000110                                         'Tell motors to go straight
   pause 200
   adcin 0, x                                               'Update variables
   adcin 1, y
   adcin 2, z
   frontwall = (((6787/(x-3)))-4)/5
   leftwall = (((6787/(y-3)))-4)/5
   flamedistance = z
   if z <= 5 then goto flame                                'Check for flame
WEND

adcin 0, x                                                  'Update variables
adcin 1, y
frontwall = (((6787/(x-3)))-4)/5
leftwall = (((6787/(y-3)))-4)/5

while leftwall < 12 and frontwall > 18                              'Tilt Right while conditions
are met
   if portc.0==0 then goto line                             'Check for line
   if linecount == 3 then goto room1                        'Check if bot need to got to room 1

   portb =%00000100                                         'Tell motors to tilt right
   pause 200
   adcin 0, x                                               'Update Variable
   adcin 1, y
   adcin 2, z
   frontwall = (((6787/(x-3)))-4)/5
   leftwall = (((6787/(y-3)))-4)/5
   flamedistance = z
   if z <= 5 then goto flame                                'Check for flame
WEND


while leftwall > 18 and frontwall > 18                              'Tilt Left while conditions are
met
   if portc.0==0 then goto line                             'Check for line
```

```
  if linecount == 3 then goto room1                    'Check if bot needs to go to room 1

  portb =%00000010                                     ' Tell motors to tilt left
  pause 200
  adcin 0, x                                           'Update variables
  adcin 1, y
  adcin 2, z
  frontwall = (((6787/(x-3)))-4)/5
  leftwall = (((6787/(y-3)))-4)/5
  flamedistance = z
  if z <= 5 then goto flame                            'Check for flame
WEND


while frontwall <= 15                                  'Turn Right while conditions are met
  if PORTC.0==0 then goto line                         'Check for line
  if linecount == 3 then goto room1                    'Check if bot needs to go to room 1
  if linecount == 5 then goto room2                    'Check if bot needs to go to room 2
  if linecount == 6 then goto room3                          'Check if bot needs to go to
room 3
  portb =%00000101                                     'Tell motors to turn right
  pause 200
  adcin 0, x                                           ' Update variables
  adcin 1, y
  adcin 2, z
  frontwall = (((6787/(x-3)))-4)/5
  leftwall = (((6787/(y-3)))-4)/5
  flamedistance = z
  if z <= 5 then goto flame                            'Check for flame

WEND

if portc.0==0 then goto line                           'Check for line
if linecount == 3 then goto room1                      'Check if bot needs to go to room 1


goto Main                                              'Start main loop again

flame:                                                 ' Flame extinguish
while flamedistance>5                                  'if flame is more than 5 cm go closer
  portb =%00000110                                     'tell motors to go straight
  pause 300
  adcin 0, x                                           'Update variables
  adcin 1, y
  adcin 2, z
  frontwall = (((6787/(x-3)))-4)/5
  leftwall = (((6787/(y-3)))-4)/5
  flamedistance = z
WEND

portb =%00010101                        ' Tell bot to spin in circles and turn
                                        ' on the fan
pause 200
portb =%00010000                        'Keep fan on during breaks in spin
pause 800
goto flame                              'Creates endless loop

Line:                                   'Line Count
linecount =  linecount+1                'Add 1 to line count
LCDout $FE, 1, #linecount               'Display line count
pause 400
goto main                               'Go back to main

turnleft:                               'Initial turn left at start of maze
pause 5000                              'Pause for 5 seconds
portb =%00001010                        'Tell motors to turn left
pause 400
linecount =  linecount+1                'Increase line count manually
goto main                               'Go back to main

turnleft1:                              'Every other turn left
portb =%00001010                        'Tell motors to turn left
pause 400
linecount =  linecount+1                'Increase line count manually
goto main                               'go back to main

room1:                                  'Program for room 1
portb =%00001001                        'Tell motors to go backwards
pause 1000
portb =%00000000                        'Reset all motors
pause 10
portb =%00000101                        'Tell motors to turn right
pause 750
portb =%00000000                        'Reset motors
goto check                              'Go to flame check procedure

room2:                                  'Program for room 2
portb =%00000101                        'turn right
```

```
pause 750
portb =%00000000
pause 10
portb=%00000110                               ‘Go straight
pause 1500
portb =%00000000
pause 10
portb =%00000101                              ‘Turn right
pause 750
portb =%00000000
goto check                                    ‘Go to flame check procedure

room3:
portb =%00000101                              ‘Turn right
pause 750
portb =%00000000
pause 10
portb=%00000110                               ‘Go straight
pause 1500
portb =%00000000
pause 10
portb =%00001010                              ‘Turn left
pause 650
portb =%00000000
pause 10

adcin 0, x                                    ‘Update variables
adcin 1, y
adcin 2, z
frontwall = (((6787/(x-3)))-4)/5
leftwall = (((6787/(y-3)))-4)/5
flamedistance = z

while frontwall>5                             ‘While conditions are met go
straight
  portb=%00000110                             ‘Go Straight
  adcin 0, x                                  ‘Update variables
  adcin 1, y
  adcin 2, z
  frontwall = (((6787/(x-3)))-4)/5
  leftwall = (((6787/(y-3)))-4)/5
  flamedistance = z
WEND
portb =%00001010                              ‘Turn left
```

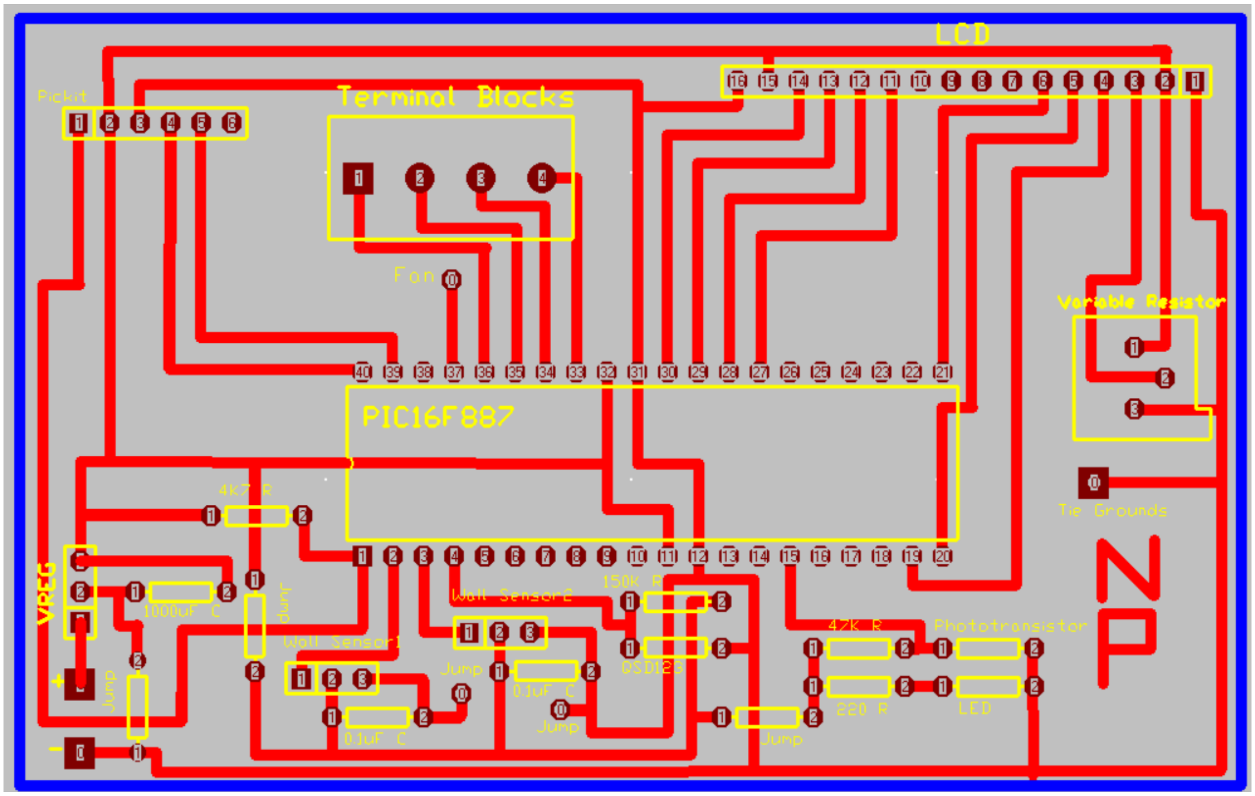
```
pause 750
portb =%00000000
pause 10
goto check                                   'Go to flame check procedure

check:                                       'Flame check procedure
adcin 0, x                                   'Update variables
adcin 1, y
adcin 2, z
frontwall = (((6787/(x-3)))-4)/5
leftwall = (((6787/(y-3)))-4)/5
flamedistance = z
pause 100
portb =%00000000                             'Reset all B Ports
if flamedistance<=50 then goto flame         'If candle less than 50cm goto flame
portb =%00000000
pause 10
if flamedistance>50 then goto turnleft1      'If no flame detected turn left
goto main                                    'Go back to main
```

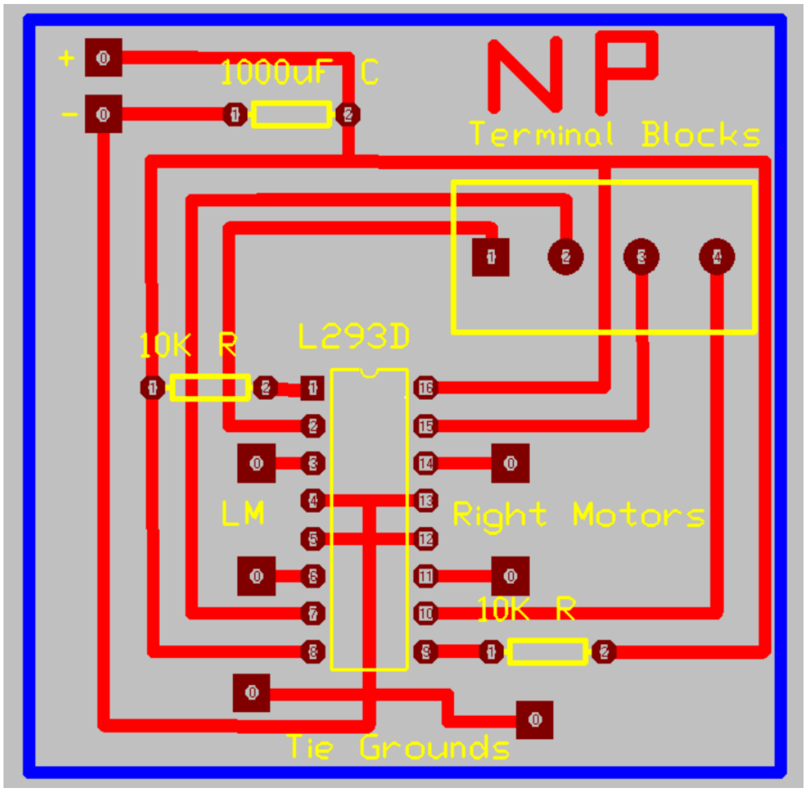
Circuit Boards:

R - Resistor

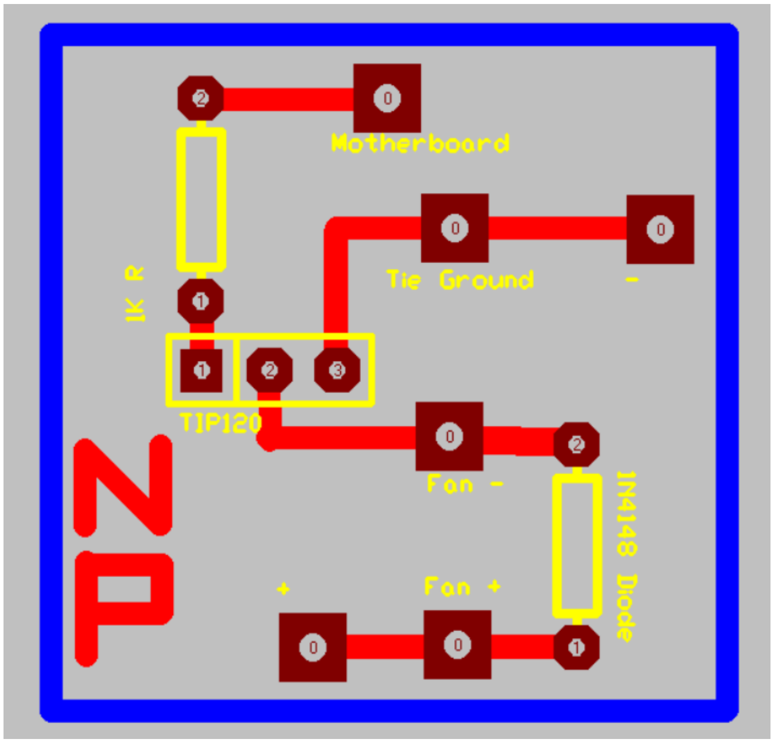
C – Capacitor



Motherboard

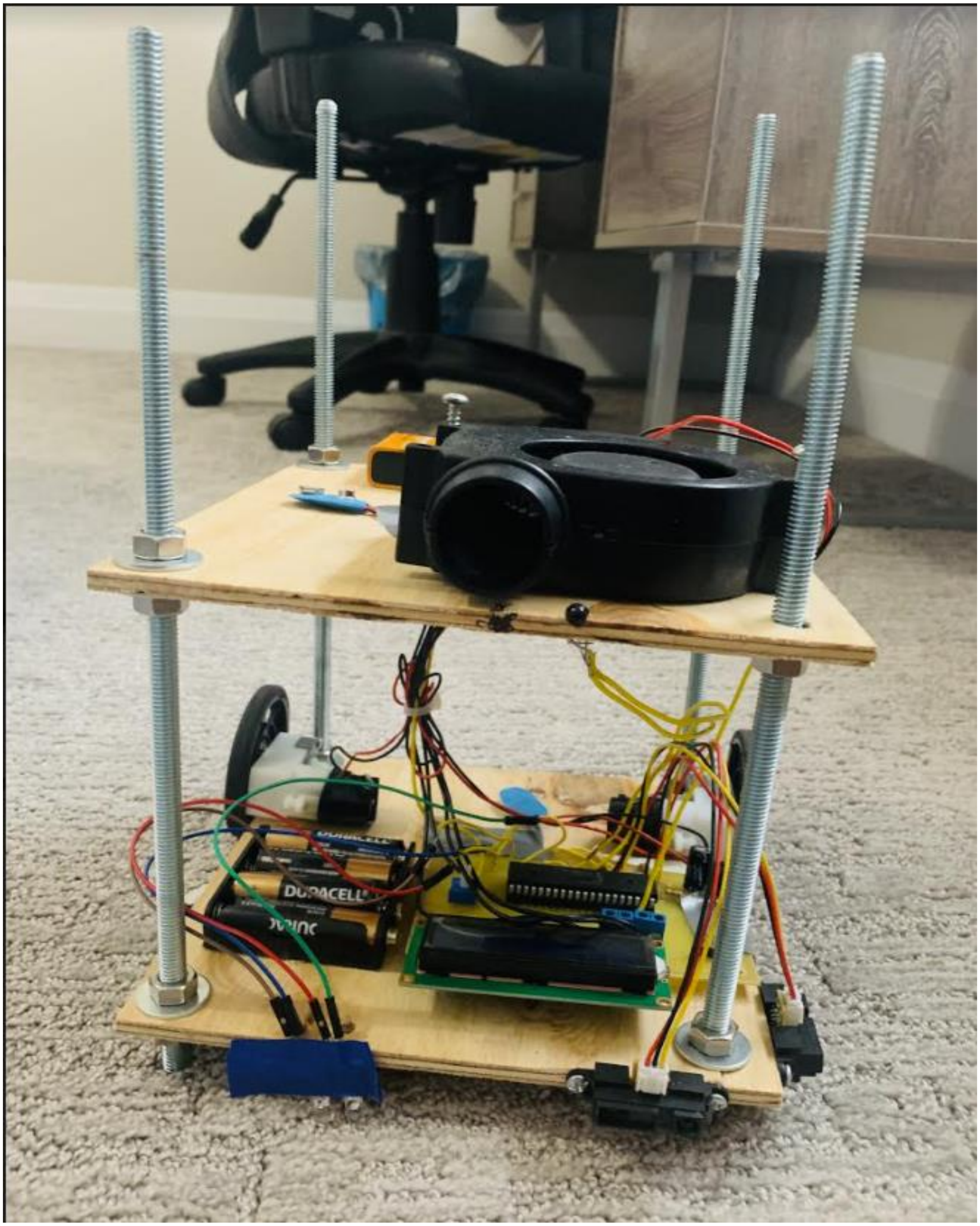


Motor Circuit

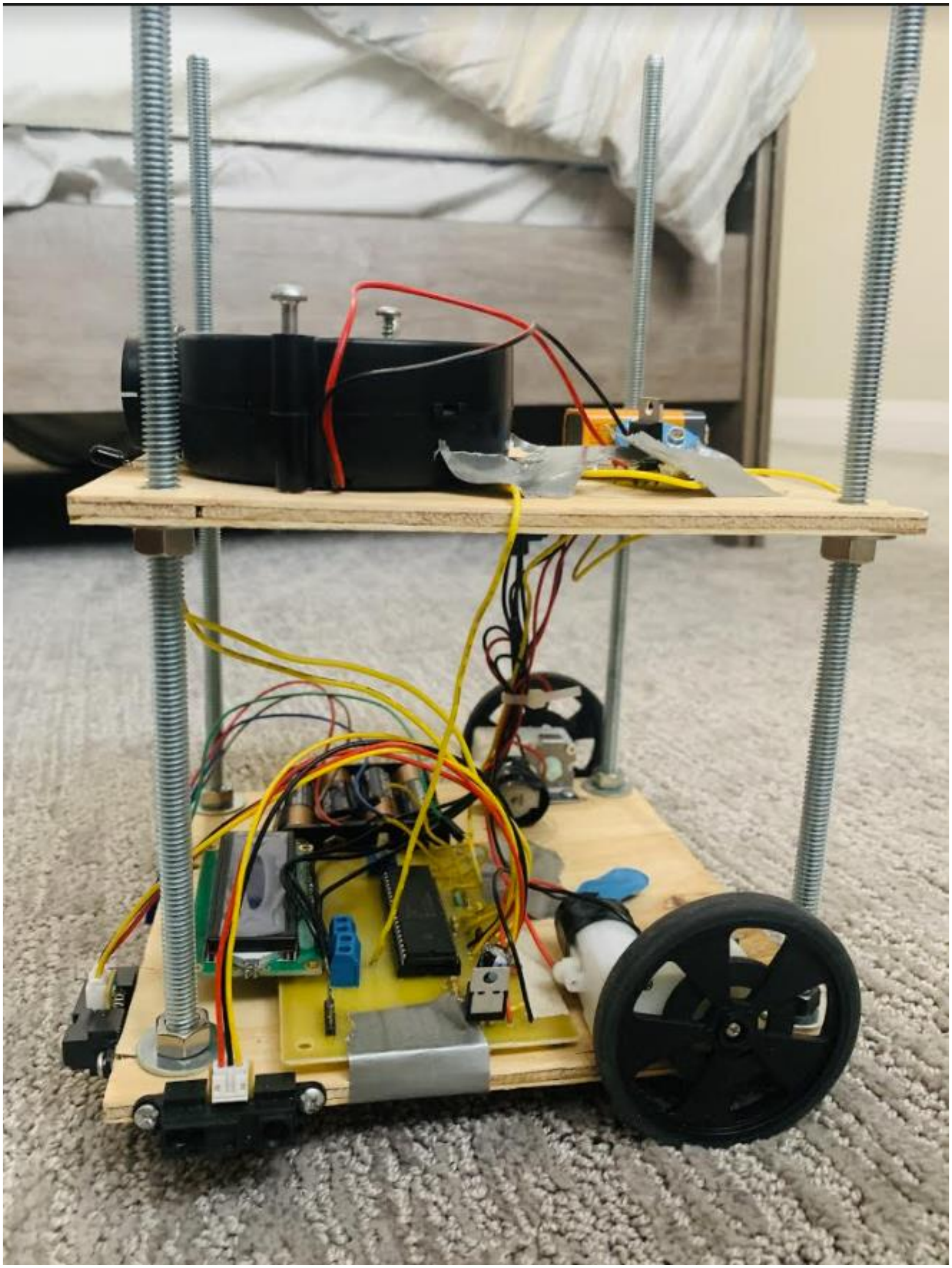


Fan Circuit

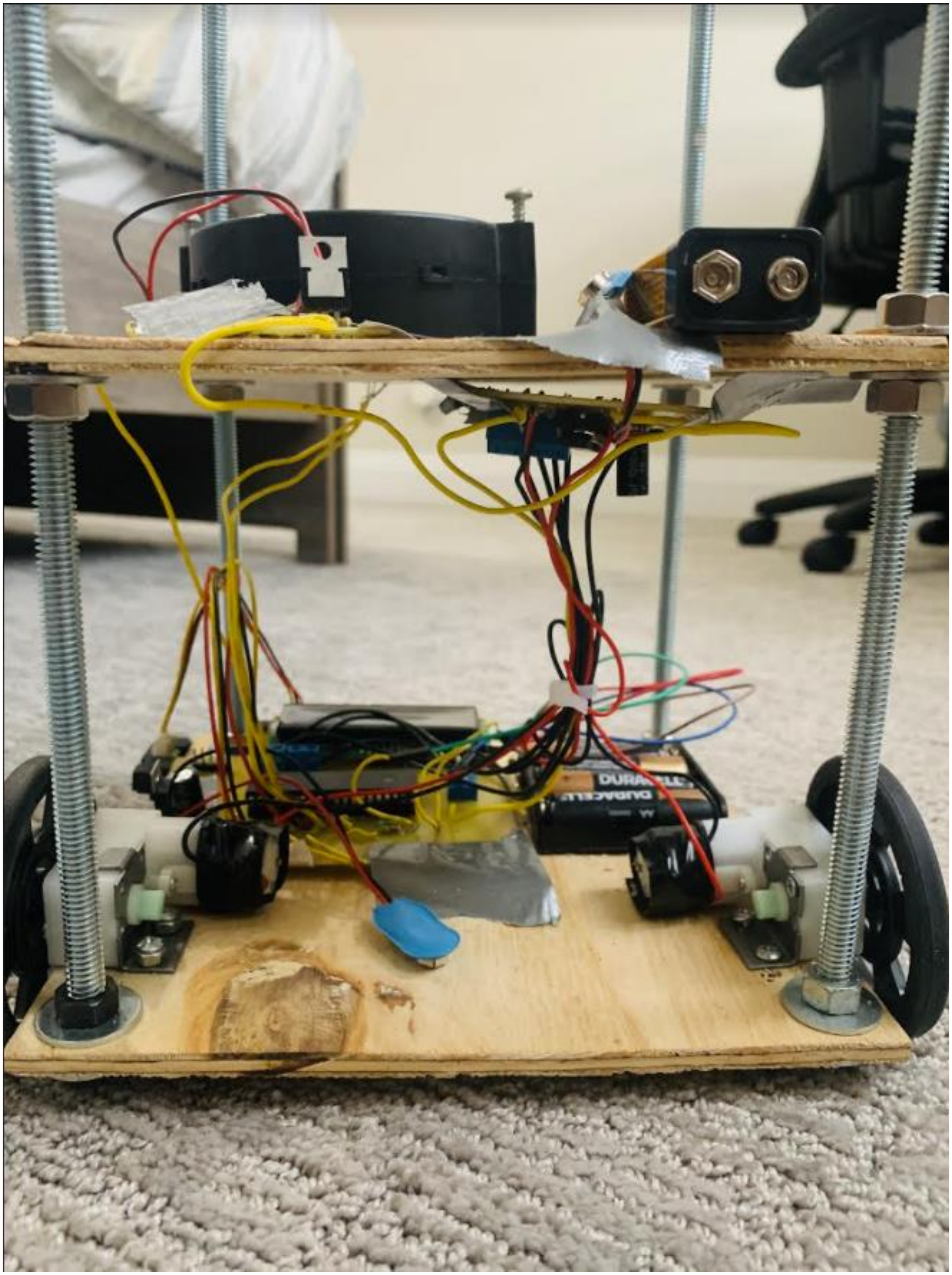
Photos:



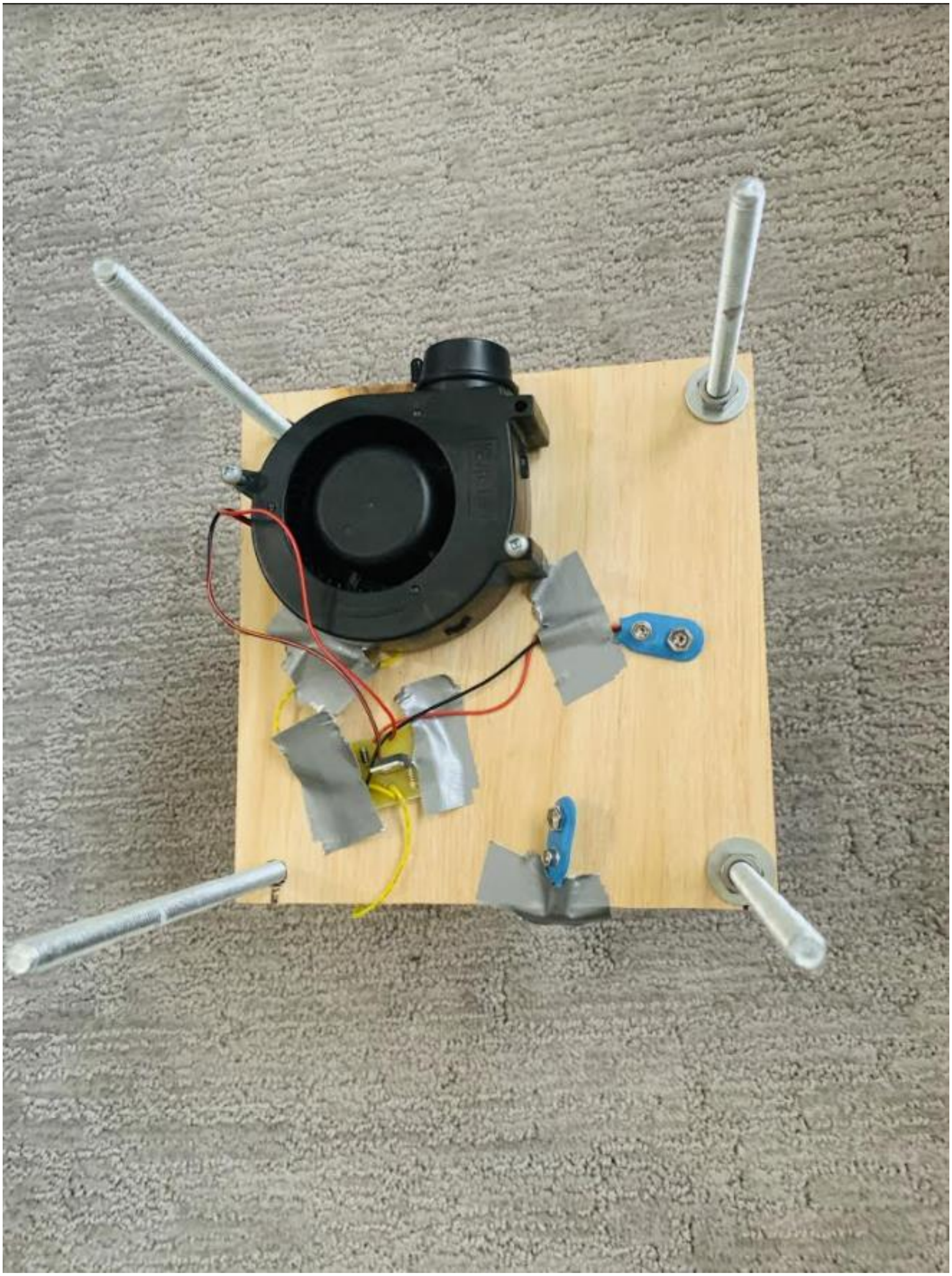
Front View



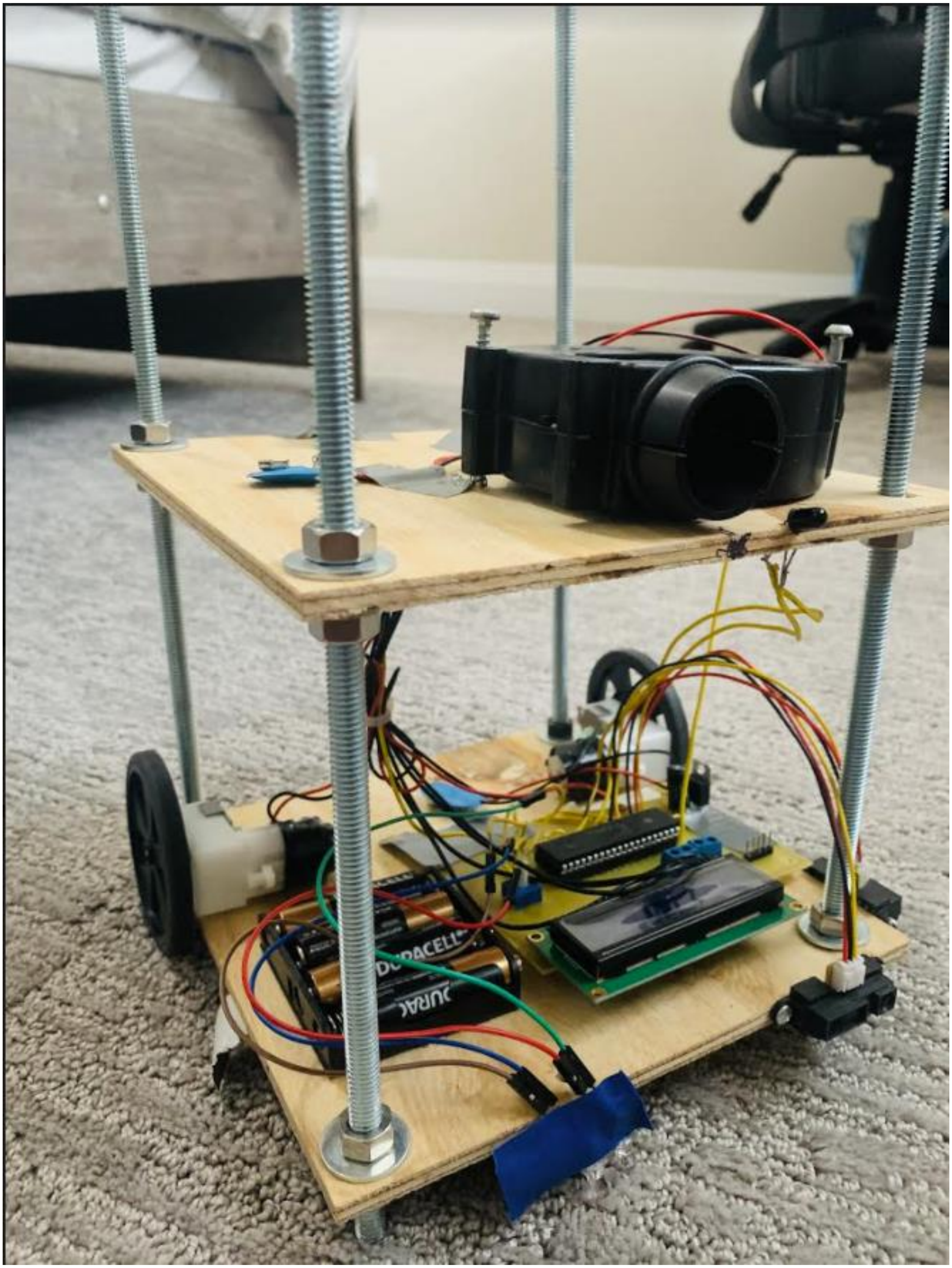
Side View



Back View



Top View



Isometric View