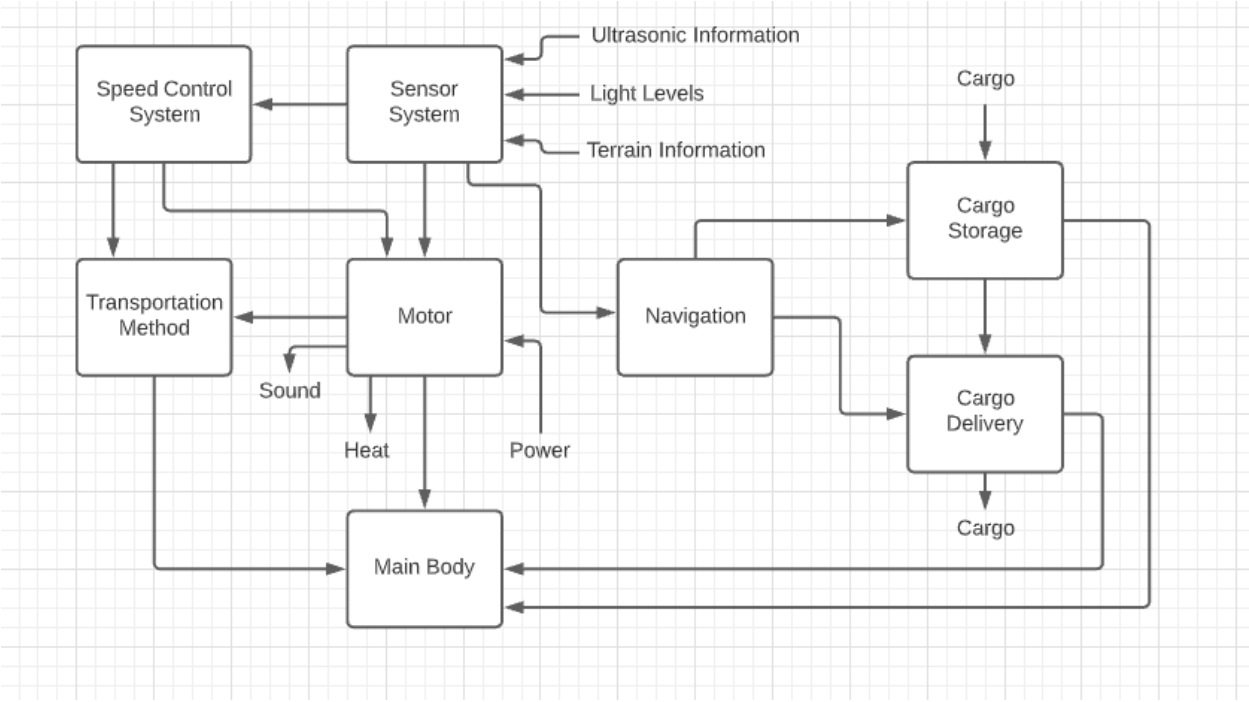
Meeting 1: 10/1/21

Meeting 1: 10/1/21 3:30-5:30

Electronic Signatures: Ella Barnes, Heather Mello, John Kang, Fahim Hossain

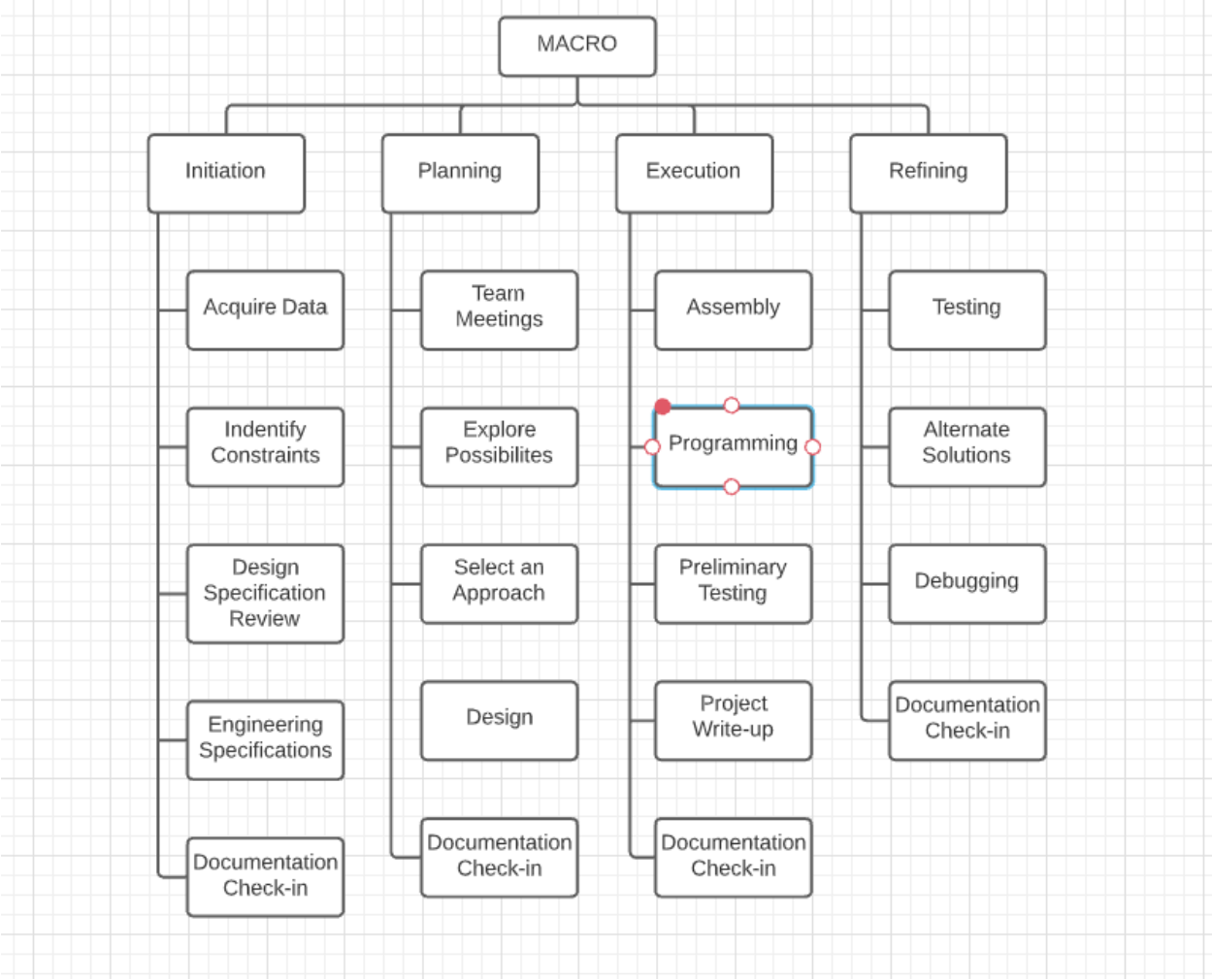
- Project 3 DSR: functional block diagram, engineering specification, work breakdown structure, gantt chart
- Batteries are corroded--remember to ask about getting new batteries next class
- Functional block diagram
    - Main blocks (inputs): cargo storage (cargo), cargo delivery, transportation method, navigation, main body, motor (power), speed control system, sensor system (ultrasonic information, light levels, terrain information)
- Engineering specification
    - 5 customer needs and technical requirements
- Work Breakdown Structure
    - 4 main categories
- Gantt Chart
    - Started, still need more information on due dates
    - Major project phases added with tasks, phases correspond to WBS categories



Speed control system: restricts the MACRO's speed to the specified maximum by limiting motor output

Transportation Method: likely wheels, how the MACRO moves around the environment

Sensor System: line finders, light sensors, IMUs, touch sensors, etc., how the the MACRO "sees" it's environment and decides how to navigate
Motor: provides power to the MACRO to allow it to move through the environment
Navigation: closely tied to the sensor system, code that takes inputs from the sensors and outputs instructions for the MACRO
Cargo Storage and Cargo Delivery: where the MACRO houses the cargo it needs to carry and delivers said cargo to the drop-off location
Main Body: chassis, houses all other components of the MACRO



Customer Need	**Technical Need**
Precise navigation to specific sites	Distance from target site
Recognition and handling hazards	Proportion of hazardous conditions that are avoided/otherwise successfully managed
Timely delivery of mission hardware	Time of delivery
Transporting cargo from location to location without dropping or tipping	Proportion of cargo that is successfully delivered
Speed restriction	Maximum speed

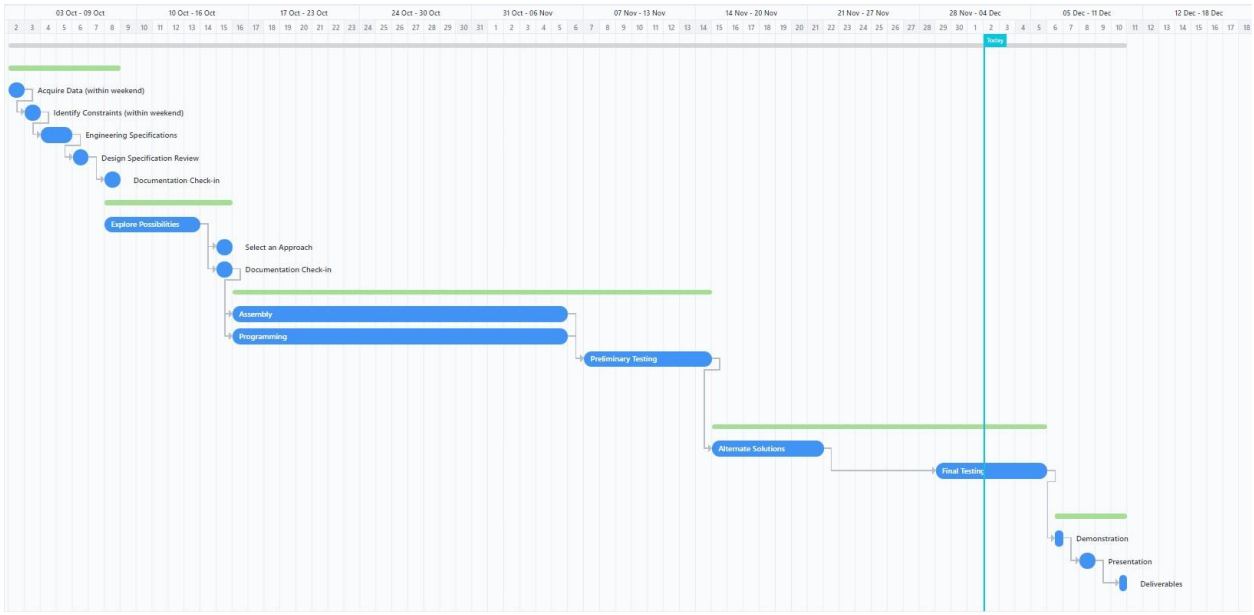
Next meeting: to be determined later today,
Outside work: none
Next meeting goal: finish gantt chart/DSR (due 10/6/21)

Meeting 2: 10/19/21
Meeting minutes: Meeting 2: 10/19/21 1:30-3:30
Electronic Signatures: Ella Barnes, John Kang, Heather Mello, Fahim Hossain

- Goal: design and build frame, accomplish task 1 of PoC
- First design: rectangular frame to hold raspberry pi
    - First iteration frame had only 4 sides with no supports, allowing it to flex and bend into diamond shapes
    - Second iteration added L-shaped supports to prevent the frame bending
- Began working on a cargo trailer design
- Redesigned/adapted the frame to fit motors

Code work from meeting 2:

```
from __future__ import print_function # use python 3 syntax but make it compatible with python 2
from __future__ import division

import time     # import the time library for the sleep function
import brickpi3 # import the BrickPi3 drivers

BP = brickpi3.BrickPi3() # Create an instance of the BrickPi3 class. BP will be the BrickPi3 object.

BP.set_motor_dps(BP.PORT_A, 0)
```



Meeting 3: 10/20/21

Meeting Minutes:

- Restructured frame around 3D printed pi frame
- Pivoted to all wheel drive, one motor for each front wheel and one motor for the back wheels
- Assembled first iteration structure
- Added a very primitive trailer
- Worked on in class tasks from the BRD lab
- Brainstormed as a group

Code work from meeting 3: none

Main parts:

1. wheels
2. cargo holder/ delivery
3. main body

wheels:

- 4 wheels on body, 2 on trailer
- treads (tank style), 2 on trailer
- 4 wheel drive
- front wheel drive
  - one motor for each front wheel
  - back wheels are just there to spin

main body:

- rectangular box
- holds pi, sensors, batteries
- holds/ pulls trailer
- wheels attached to bottom

cargo holder / delivery:

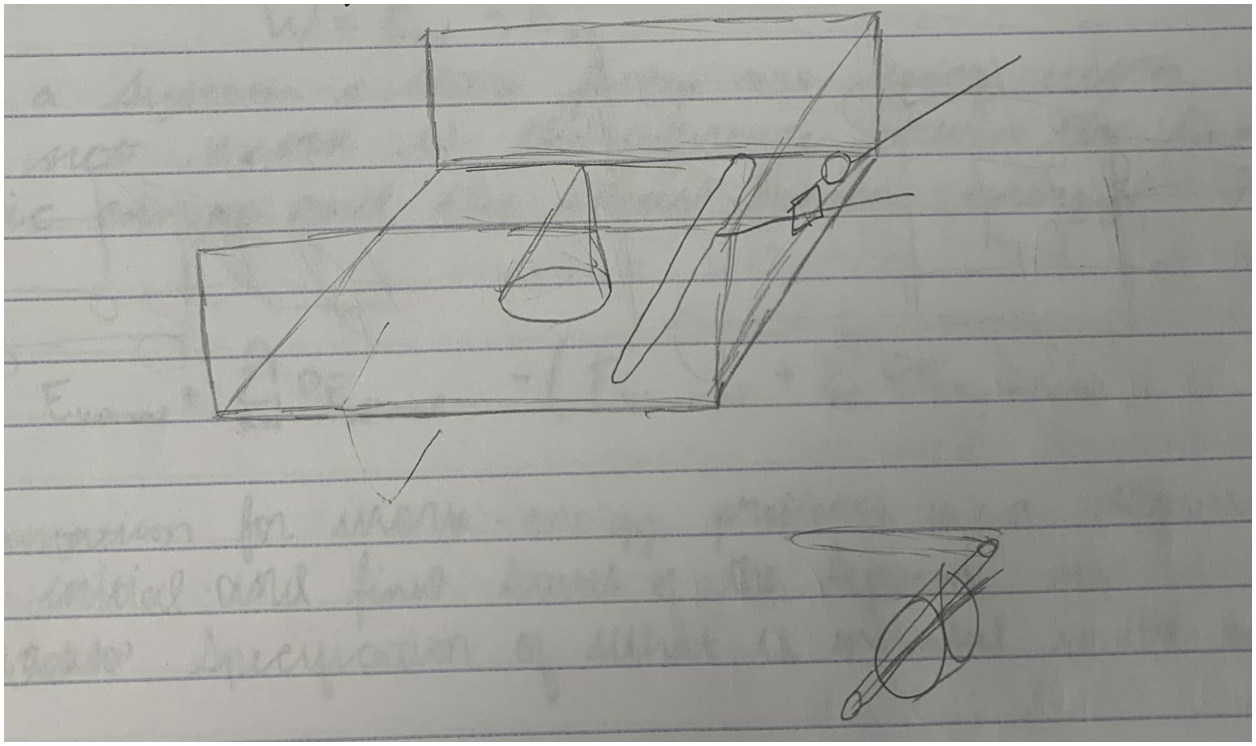
- floor tips to an angle, cargo slides out
- arm picks it up & drops it
- floor slides out from underneath
- separate trailer attached loosely
- bingo wheel
- cargo trailer super low to the ground
  - ↳ arm sweeps cargo out of the trailer

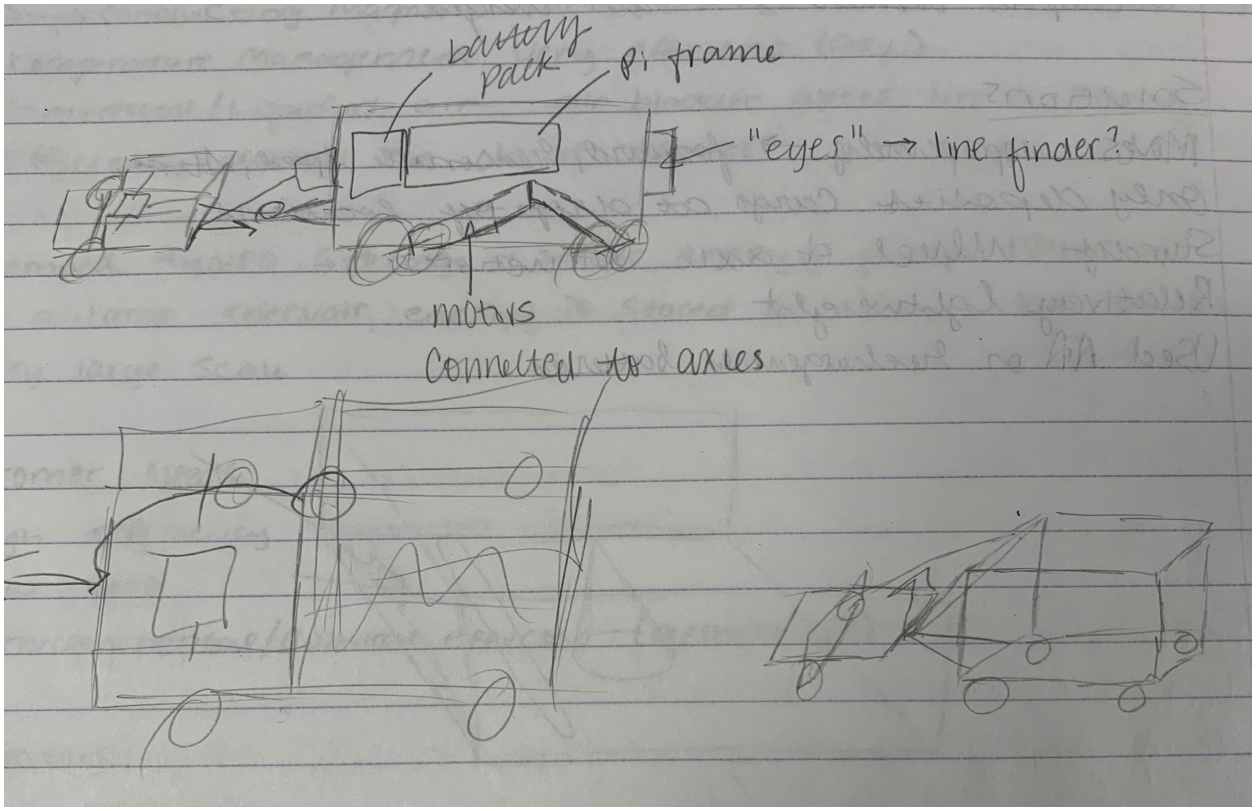
Worst Possible Idea:

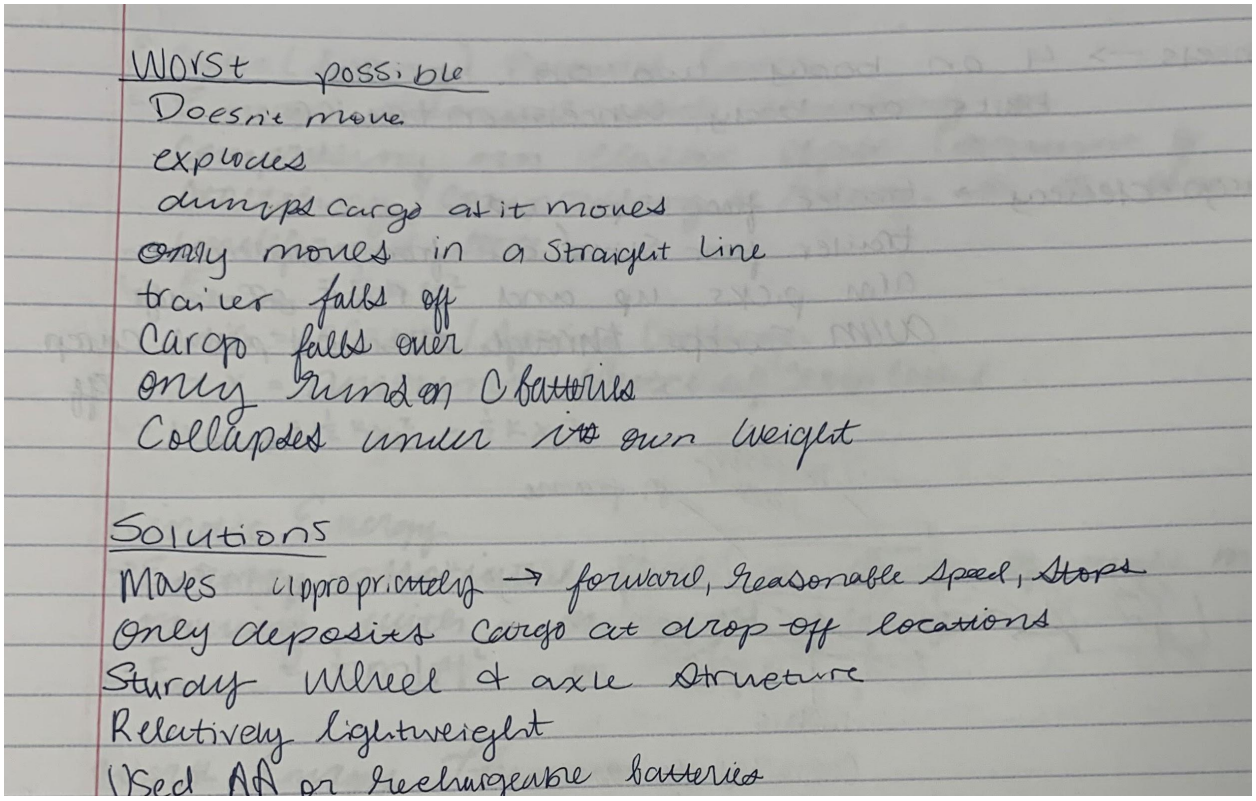
- ✓dumps cargo as it goes ~~explodes~~
- ✓only moves backwards
- ✓wheels fall off
- ✓cargo falls over
- ✓only follows straight solid lines
- ✓trailer falls off
- ✓has to be plugged in
- ✓goes way too fast
- ✓moves too slow
- ✓can't go over any hills
- ✓sudo poweroff when scared
- ✓has emotions
- ✓never stops

Solutions:

- moves forwards at a reasonable speed
- not sentient
- stops when it needs to
- sturdy structure
- caps on the end of axles
- relatively lightweight
- runs off of AA batteries or rechargable
- holds cargo securely
- sensors allow it to follow different types of lines
- front wheel drive
- one motor for each side




battery pack
Pi frame
"eyes" → line finder?
motors
Connected to axles



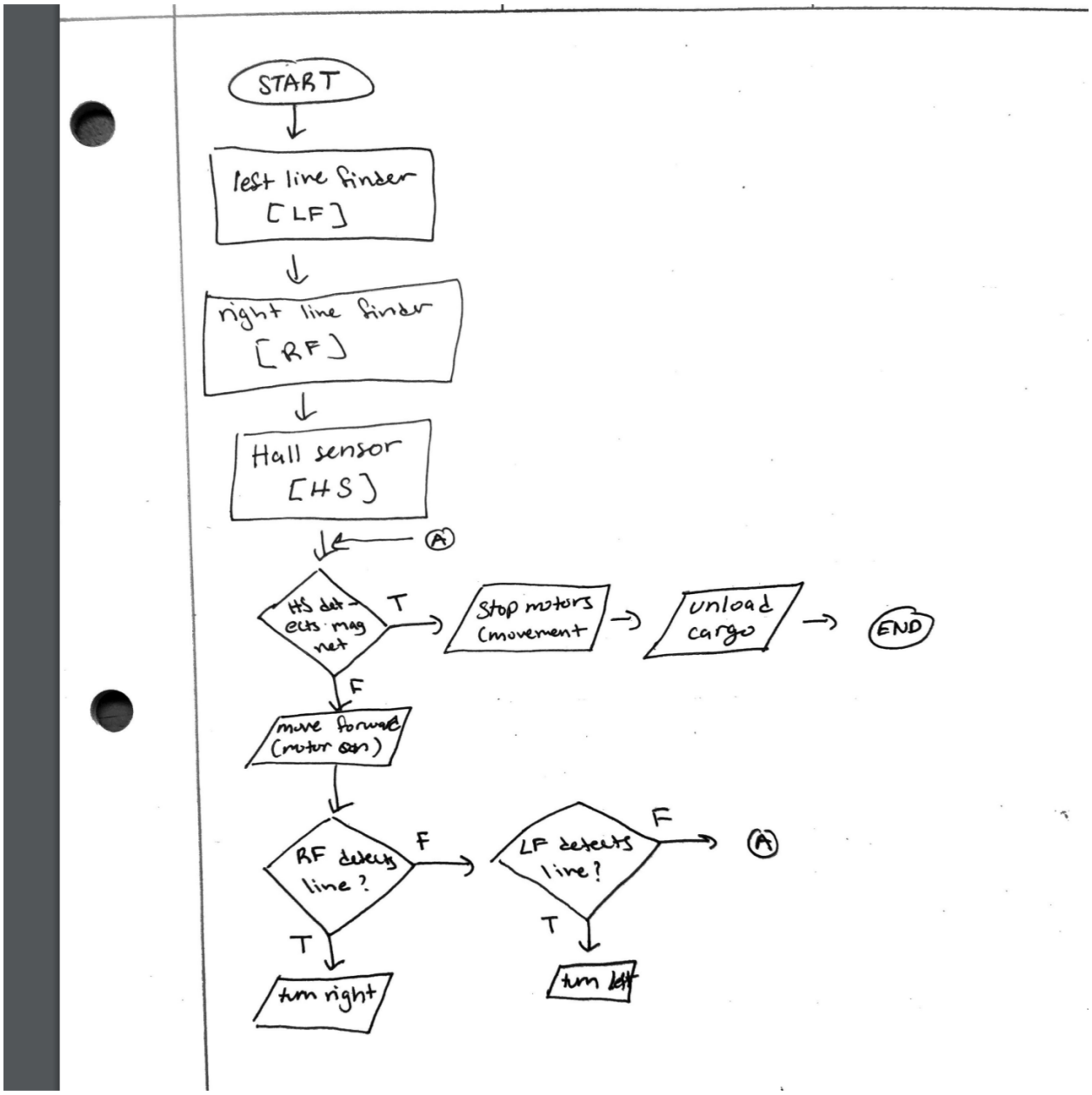
Worst possible
Doesn't move
explodes
dumps cargo as it moves
only moves in a straight line
trailer falls off
Cargo falls over
only runs on C batteries
Collapses under its own weight

Solutions
Moves appropriately → forward, reasonable speed, stops
Only deposits cargo at drop off locations
Sturdy wheel & axle structure
Relatively lightweight
Used AA or rechargeable batteries

Meeting 4: 10/21/21 2:30-4:00

Electronic Signatures: Ella Barnes, Heather Mello, John Kang, Fahim Hossain

- Added a “basket” to hold the battery pack below the main body of the robot
- Redesigned back wheels (removed motor)
- Completed first iteration robot
- Came up with rough flowchart for how the robot would operate

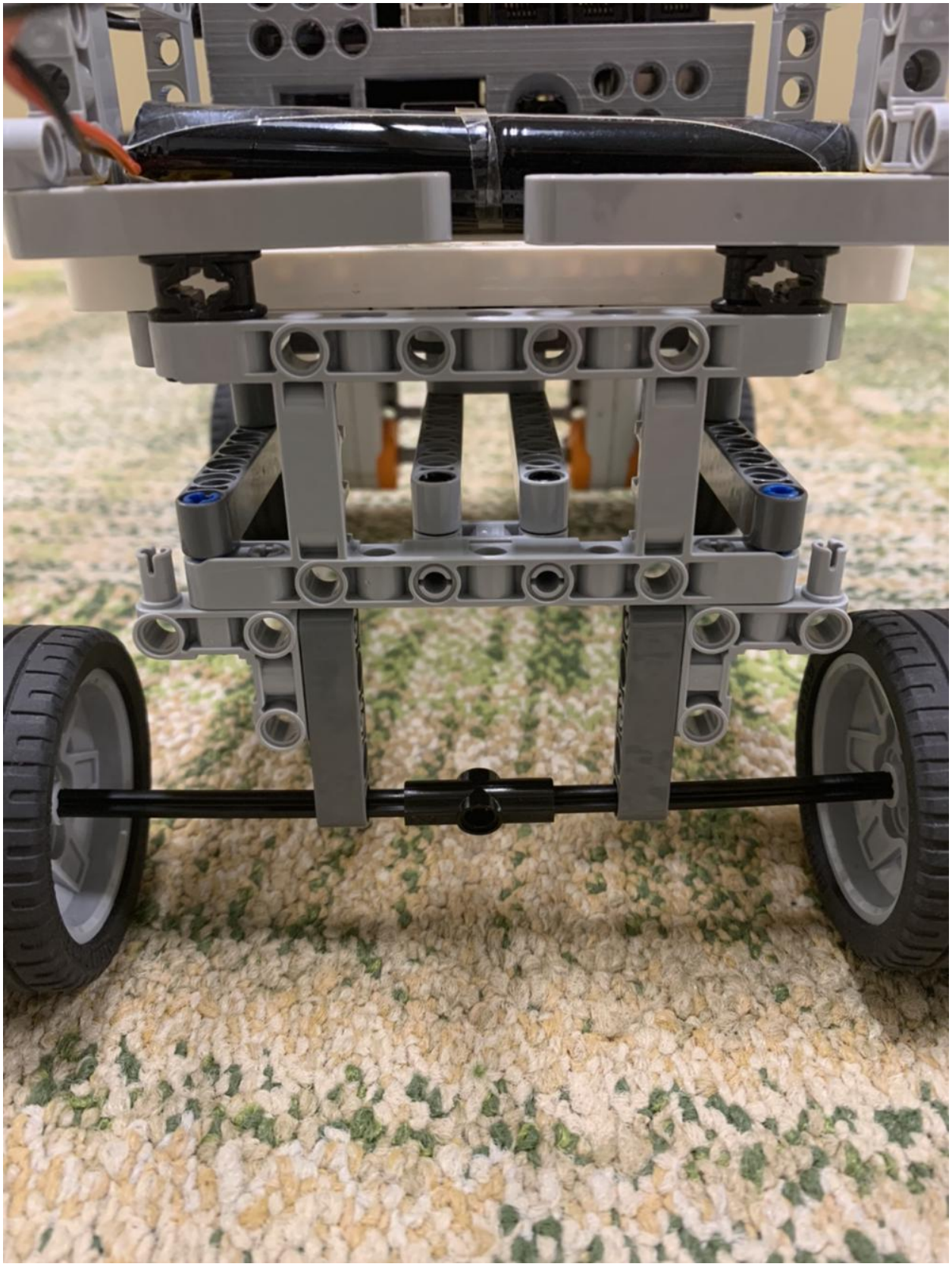


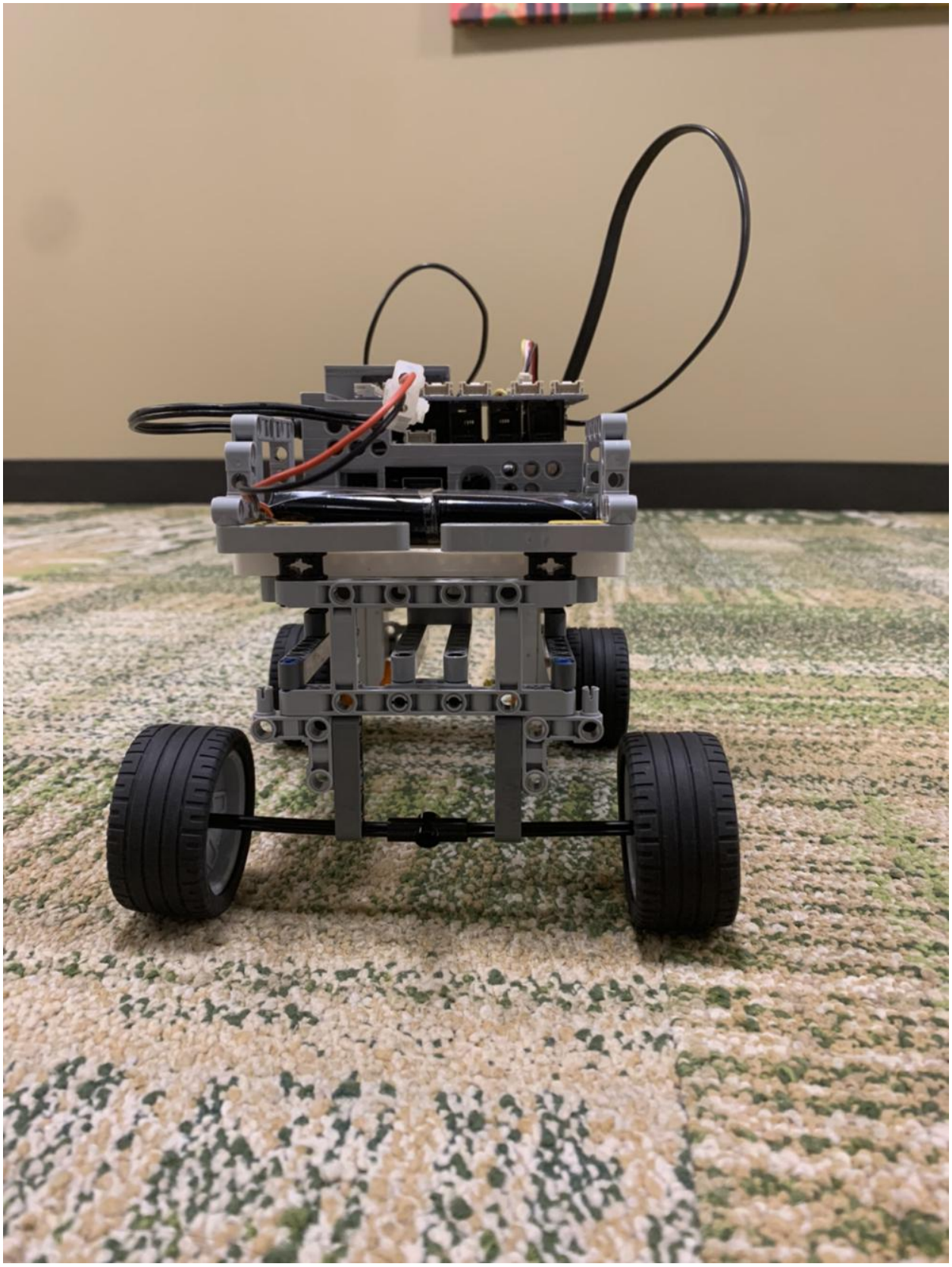
Meeting 5: 10/21/21 8:00-10:00pm
Electronic Signatures: Ella Barnes, Heather Mello, John Kang, Fahim Hossain

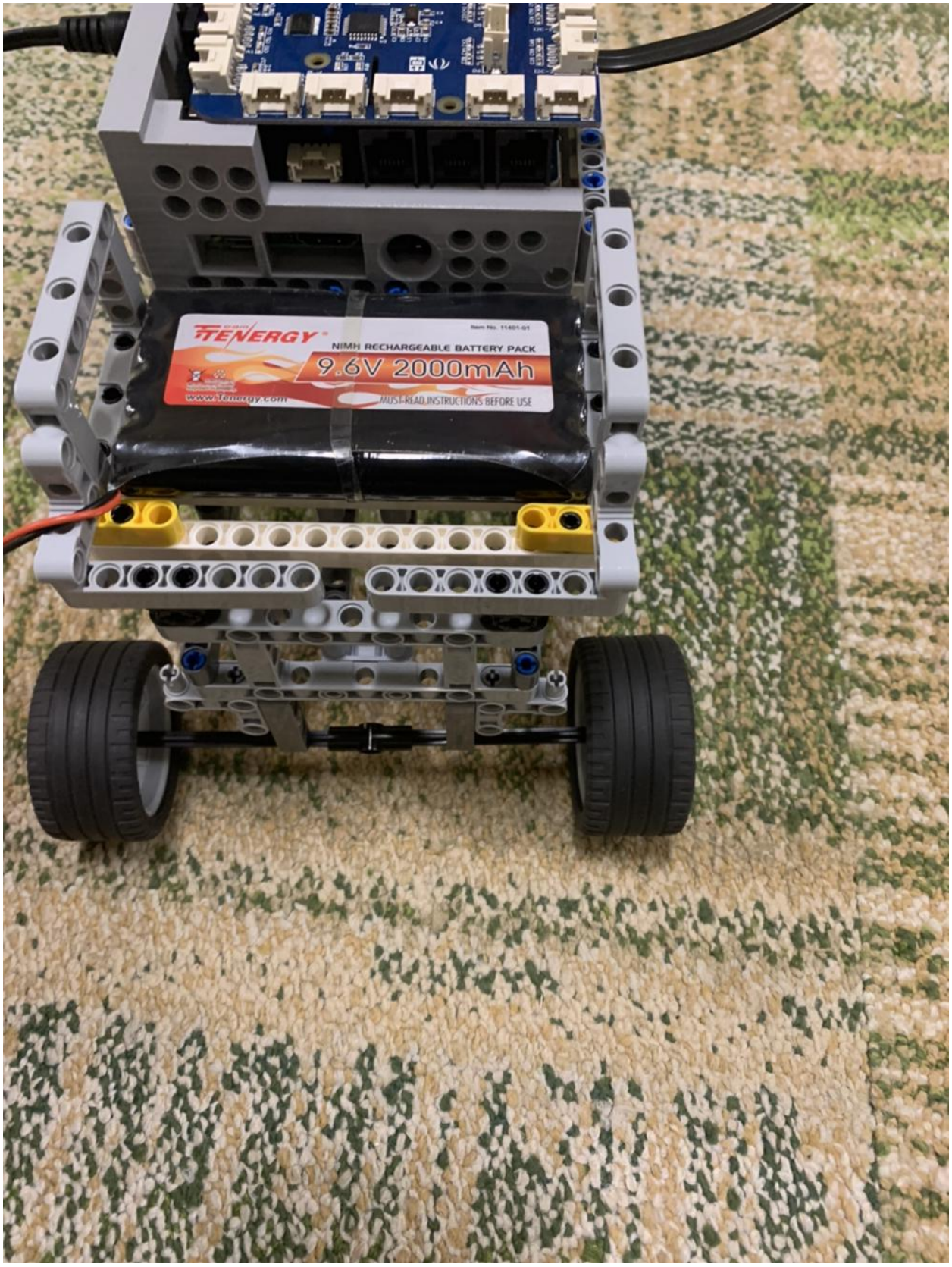
- Realized pi1 does not work as expected--need to mention in class tomorrow
    - Pi1 grovepi is believed to be the source of the issues as brickpi 1 is able to power motors but sensors do not function
- Modified battery basket to be modular and able to hold the rechargeable batteries
- Expected results of in-class test 10/22/21: robot is expected to be able to move forward but not turn and is expected to be able to follow a straight black line and stop moving when the line stops
- Line finders are taped to the body with painter's tape
- Front wheels are sagging--likely due to excess weight in the center of the frame from the batteries and pi complex
- Pi3 froze up and had to be turned off without shutting down in terminal--unknown precisely what the consequences of this will be. After turning pi3 off with the switch, nothing happened (i.e. the problem was not resolved), so it was unplugged. The pi was plugged into the wall and turned on and worked normally--the non-rechargeable batteries are now believed to be the source of the issue
- Upon further testing plugged into the wall, the line finders do not work very well. They are simultaneously too sensitive and not sensitive enough

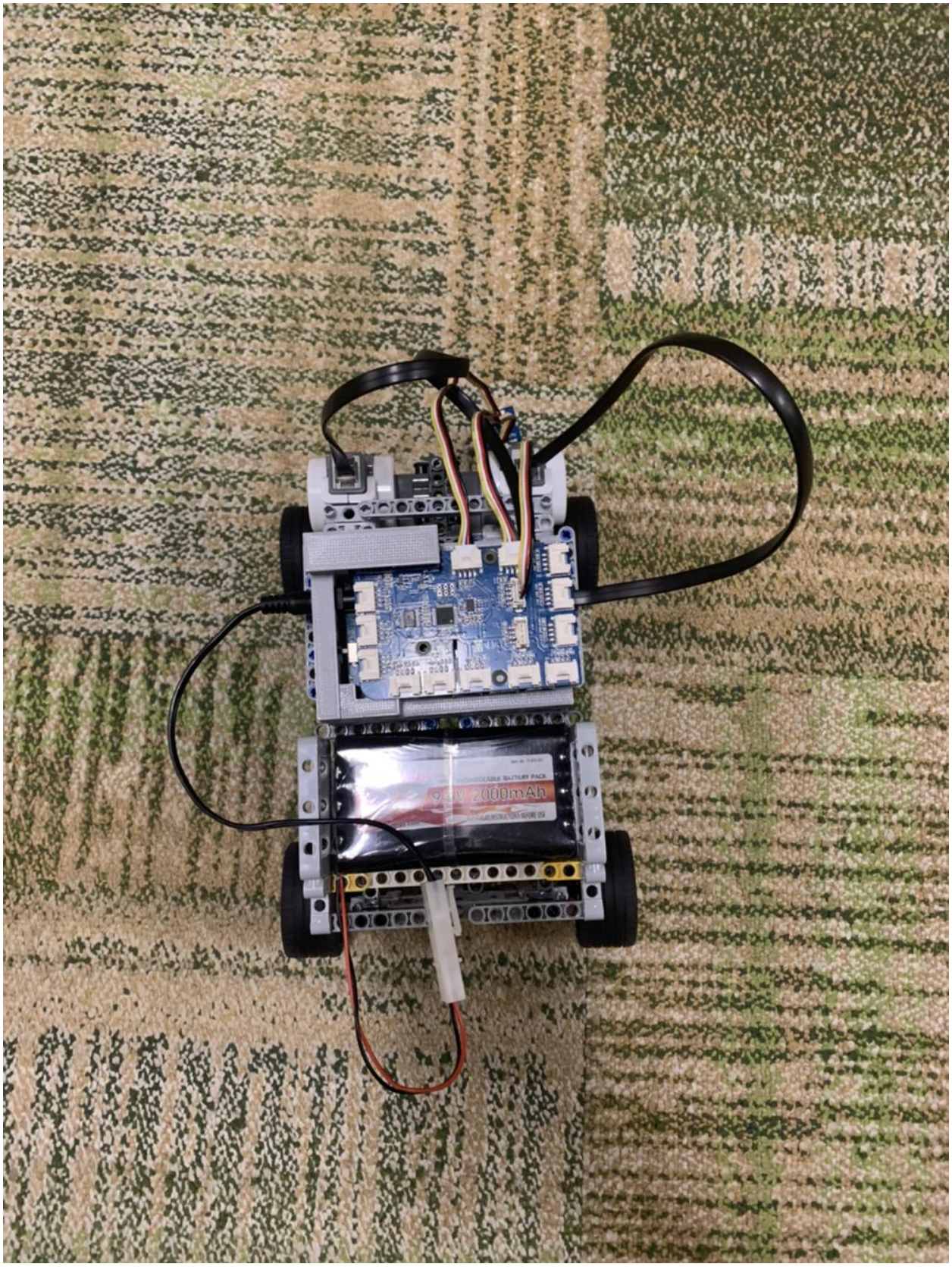
Meeting 6: 10/28/21 2:30-4:30
Electronic Signatures: Ella Barnes, Heather Mello, John Kang

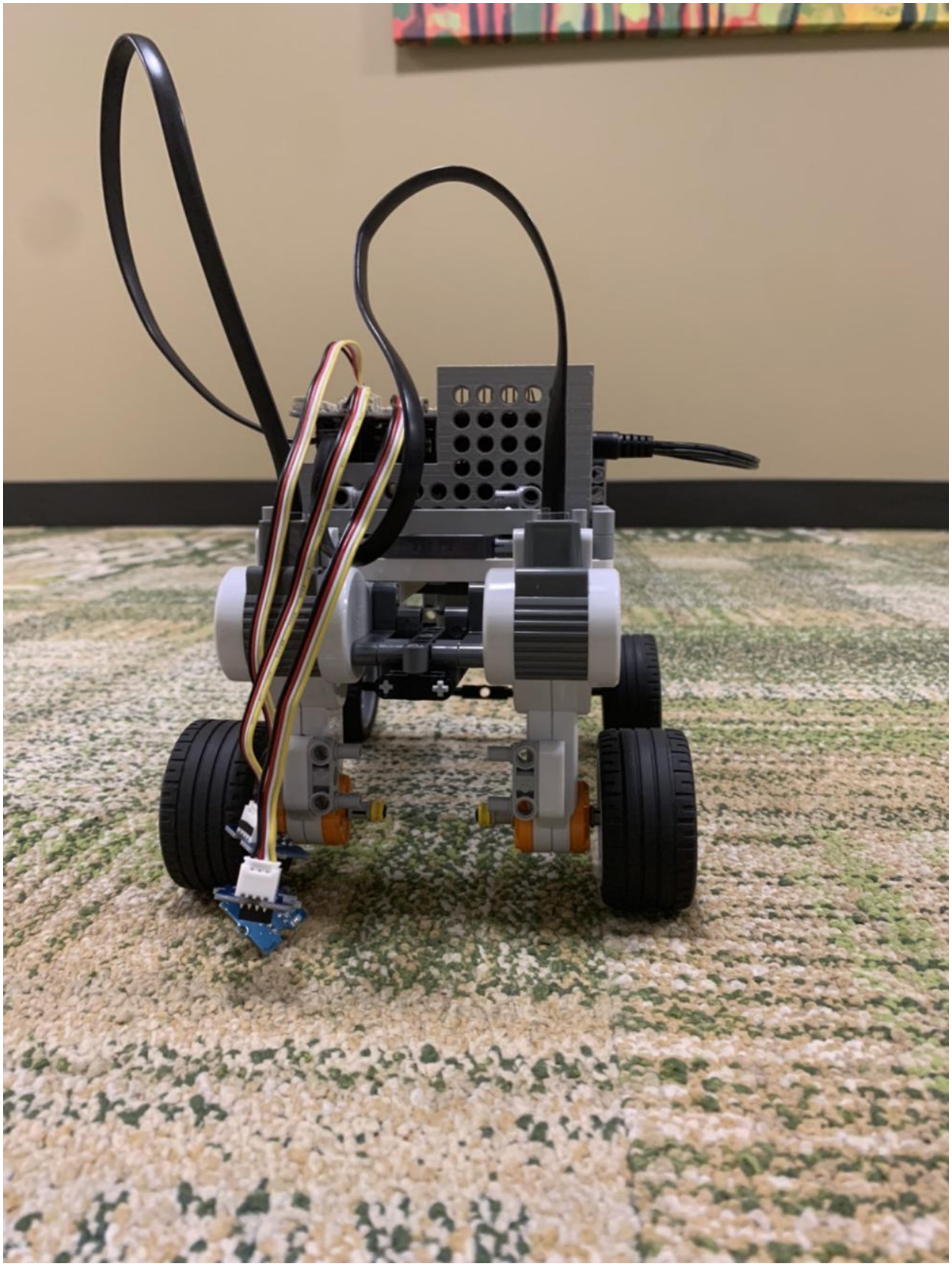
- Rebuilt the body of the robot
    - Made it longer, kept width the same
- Rebuilt the trailer
    - Wider to accommodate the size of cargo
- Images:

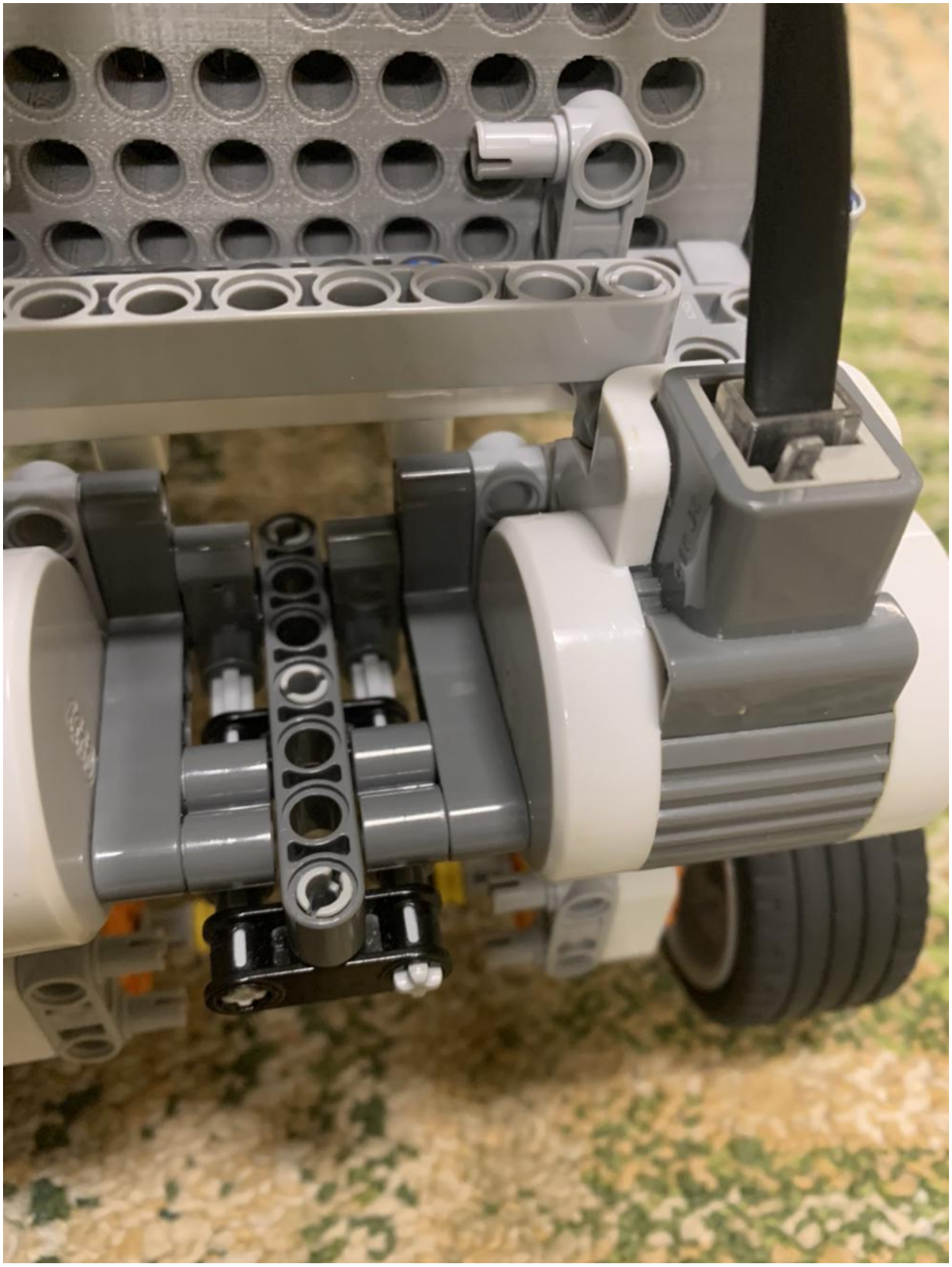


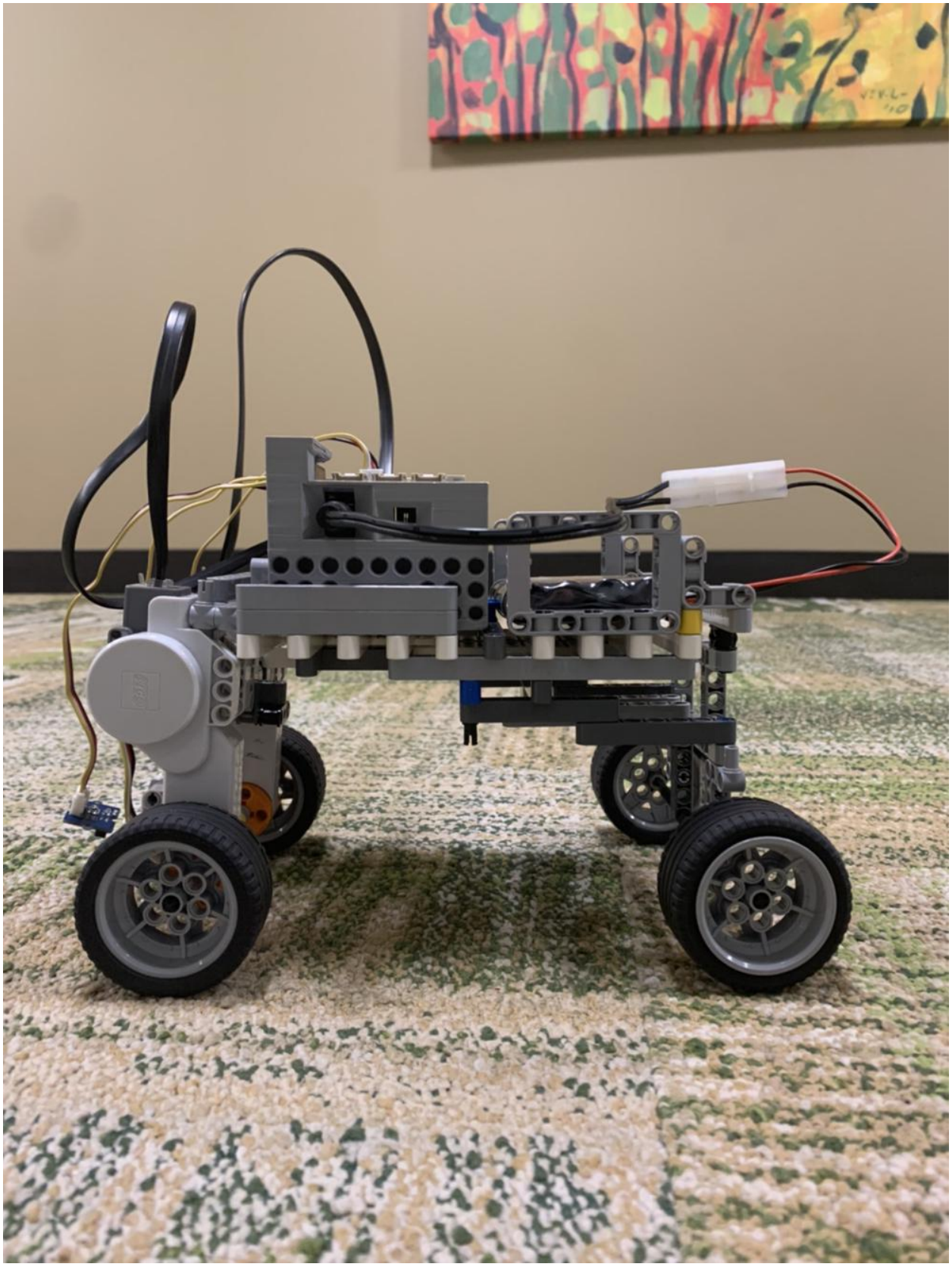


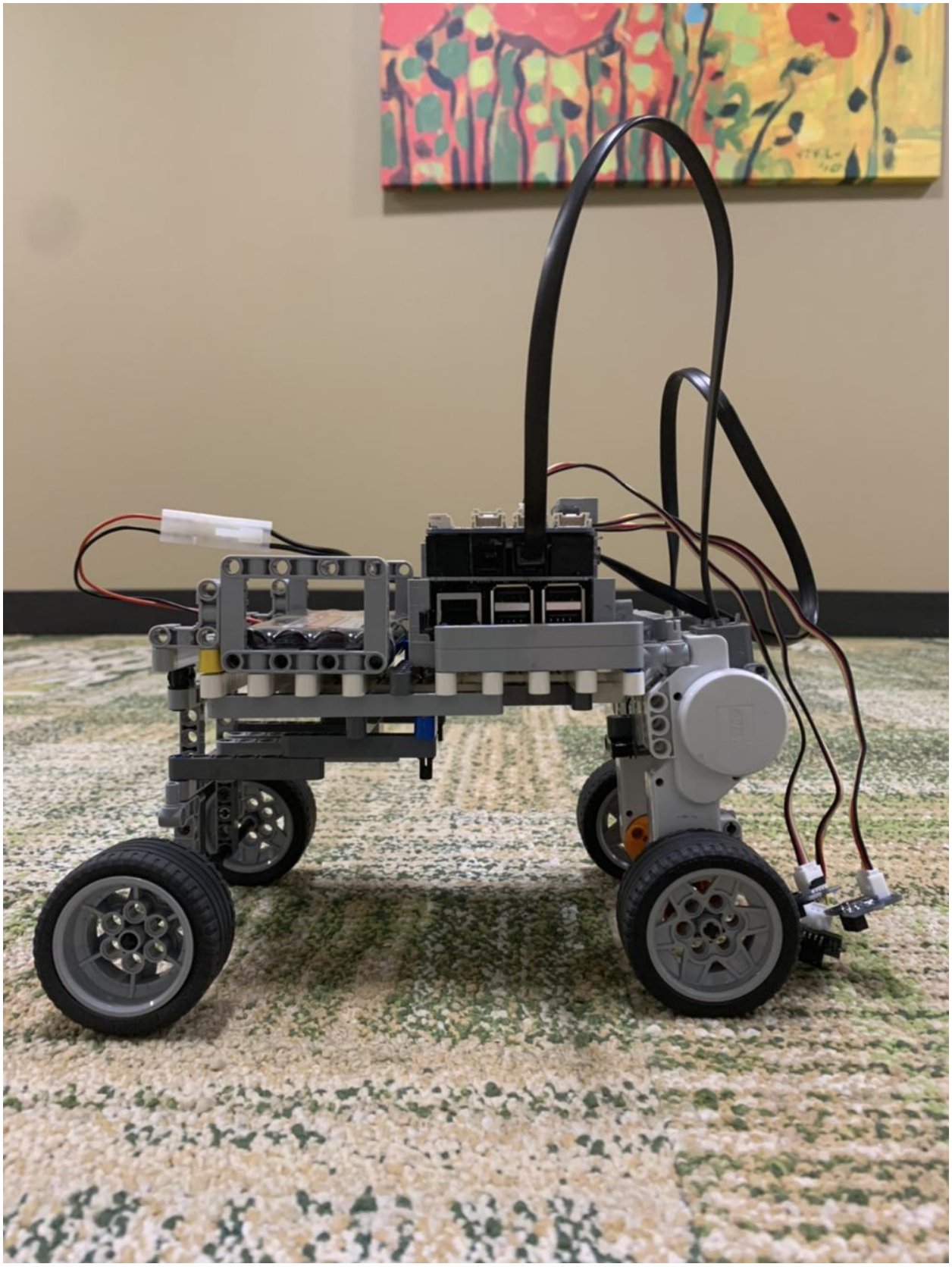

TENERGY
NIMH RECHARGEABLE BATTERY PACK
9.6V 2000mAh
www.Tenergy.com
MUST-READ INSTRUCTIONS BEFORE USE


9.6V 2000mAh







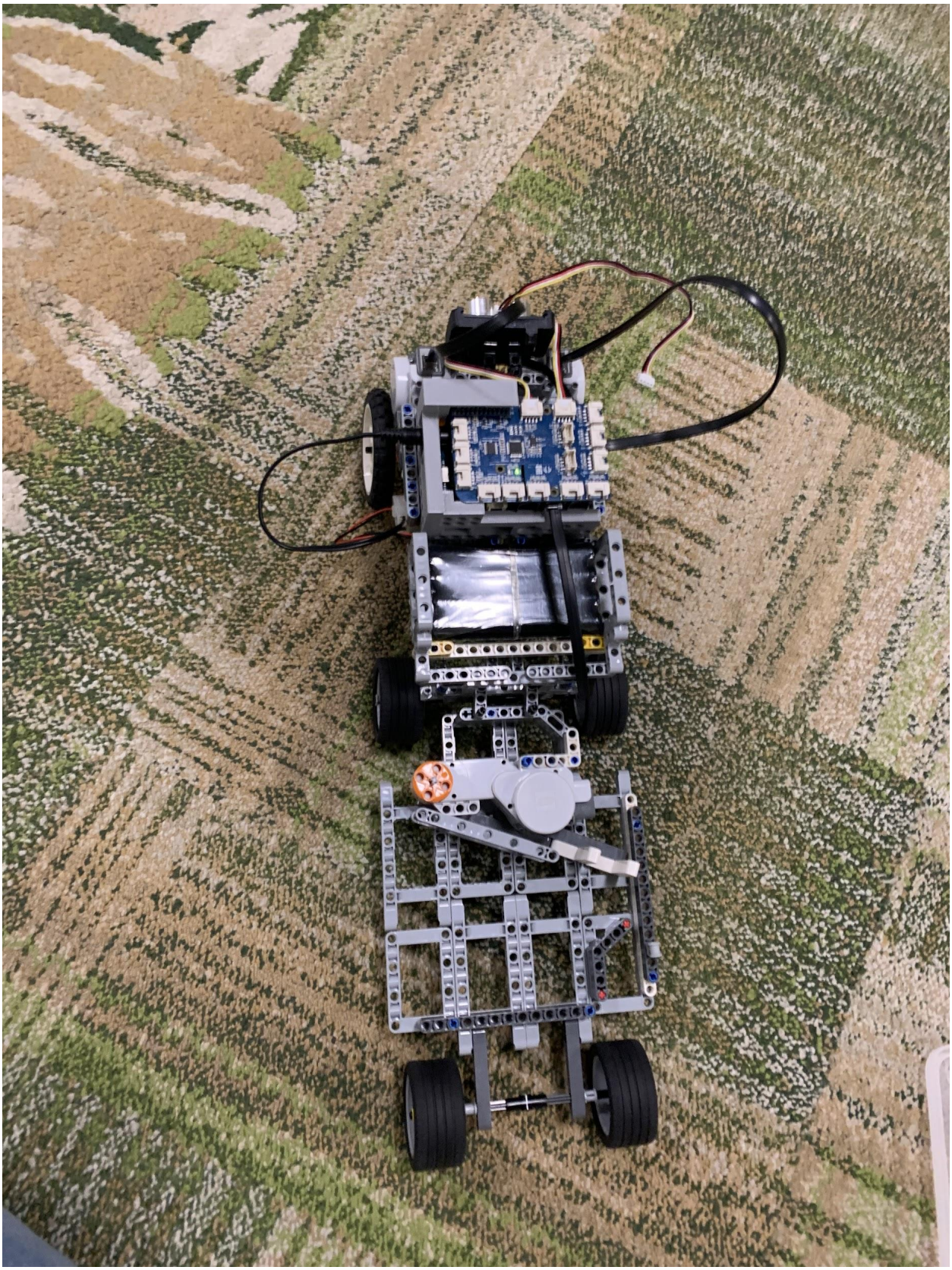


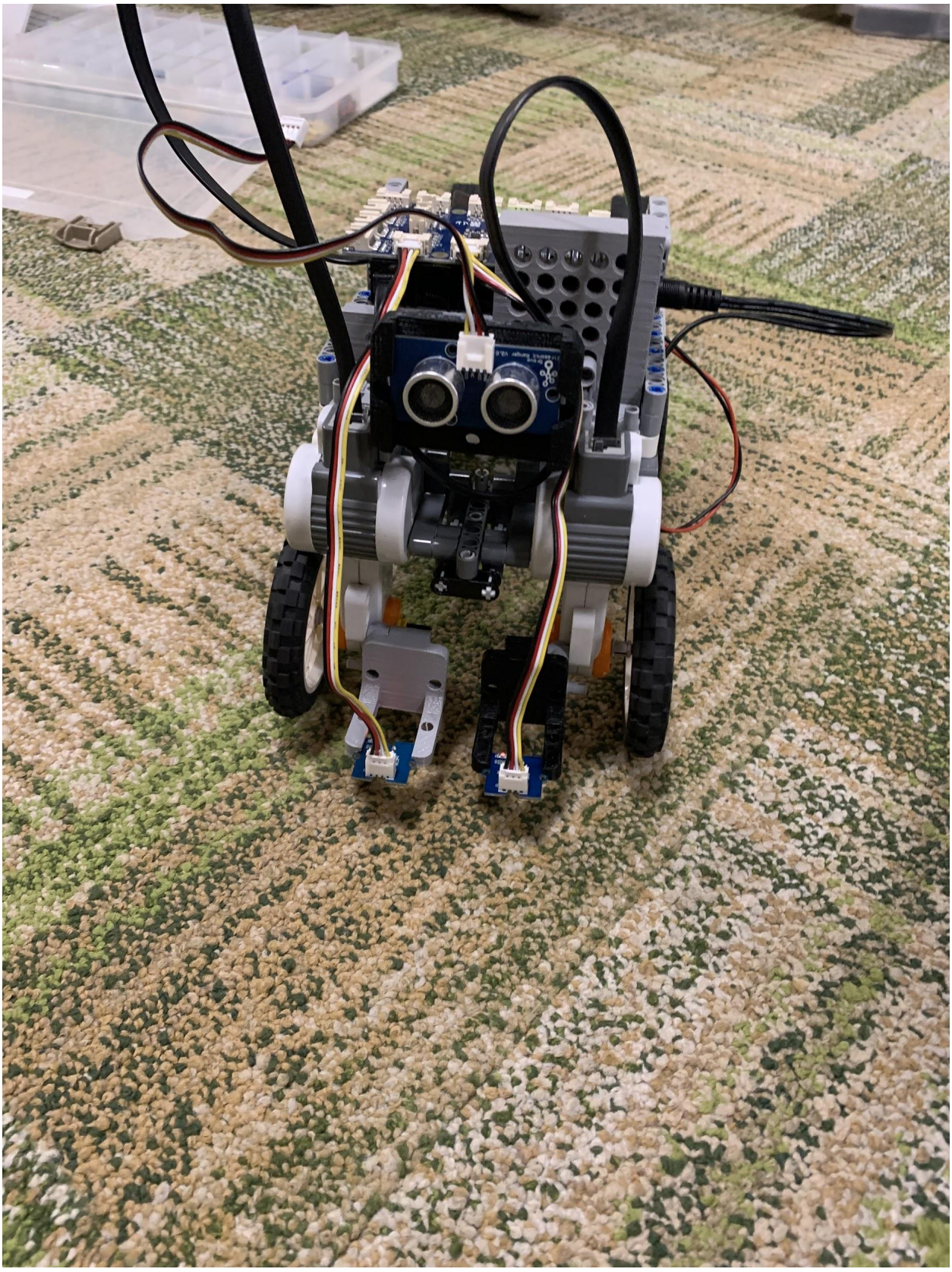
Meeting 7: 11/11/21 2:45-4:00
Electronic Signatures: Heather Mello, Ella Barnes, John Kang

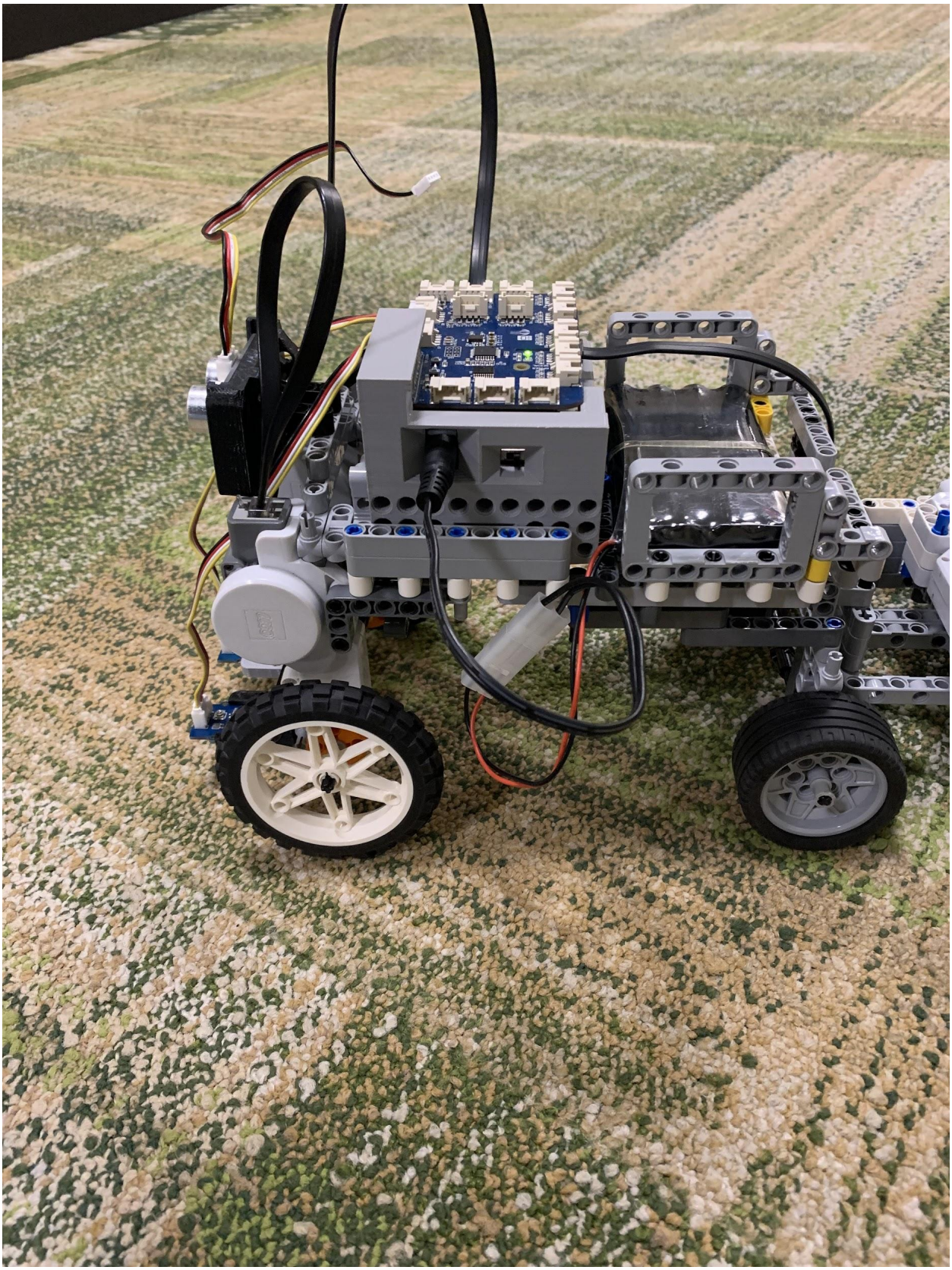
- Swapped out wheels:
    - Front wheels are now ‘motorcycle wheels’
    - Back wheels are same as before (medium wheels), but without rubber
    - Trailer wheels are now medium wheels
- Strengthened the attachment of the motors to the body
- Tested the code and confirmed that it worked with new wheels on robot
- Added line finders to front bottom of main body, by the wheels

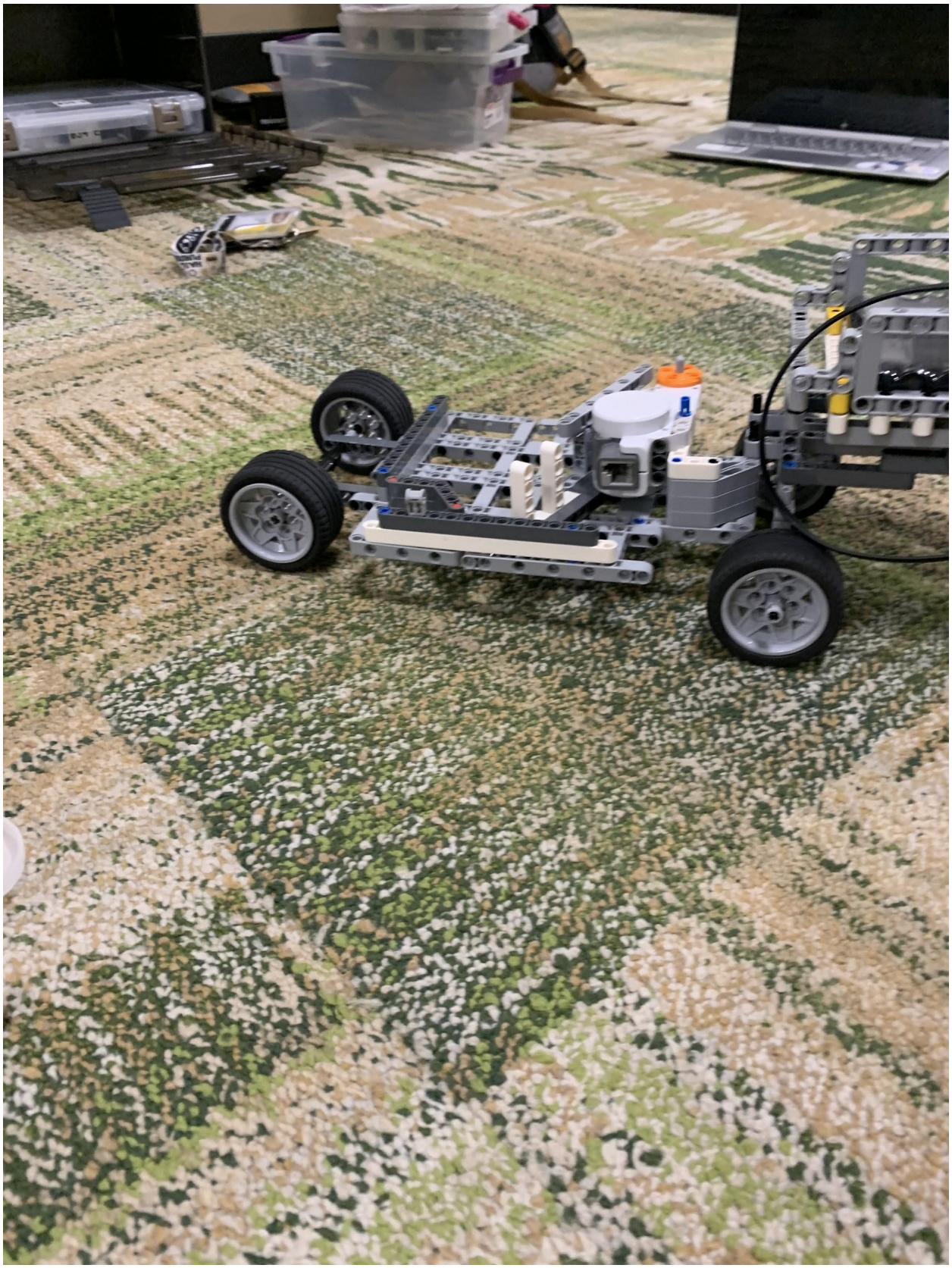
Meeting 8: 11/14/21 1:00-2:30
Electronic Signatures: Ella Barnes, Heather Mello, John Kang

- Modified trailer design to have two attachments
- Added ultrasonic sensor to front of main body
- Ran out of small connectors (tragic)
- Further modified trailer design to remove the trailer's swiveling abilities in favor of a more stable connection that is less likely to pull itself apart when weight is added
- Designed trailer motor system
    - Need further reinforcement for trailer arm
- Attended office hours to test code and robot's ability to follow lines, broken lines, and trailer's ability to push cargo off
    - Robot was incapable of following a track, possibly due to line finder issues
- Images









Meeting 9: 11/18/21 2:00-3:30pm
Electronic Signatures: Ella Barnes, John Kang, Heather Mello, Fahim Hossain

- Redesigned trailer to better fit cargo
- Tested sensors
    - Code was not working at last in class activity
    - Determined the problem was due to the sensors, made plans to go to office hours on Saturday, November 20, for assistance
- Added smaller wheels to trailer for extra support under cargo weight
    - Two sets, one larger than the other
    - Larger set of auxiliary wheels was added to the middle of the trailer, smaller set was added to the front

Meeting 10: 11/20/21
Electronic Signatures: Ella Barnes, Fahim Hossain, John Kang, Heather Mello

- Office hours
    - Worked on line finder calibration and tested current design

Meeting 11: 11/21/21
Electronic Signatures: Ella Barnes, Fahim Hossain, John Kang, Heather Mello

- Tested sensors
    - Sensors seemed to be working as expected
- Modified trailer design
    - Added an upper section to the cargo arm for added strength
    - Added legs to the base of the trailer to prevent it from sagging under the weight of the cargo
    - Added a support to the motor to keep the arm from falling off if it went too far over the edge of the trailer

CODE:

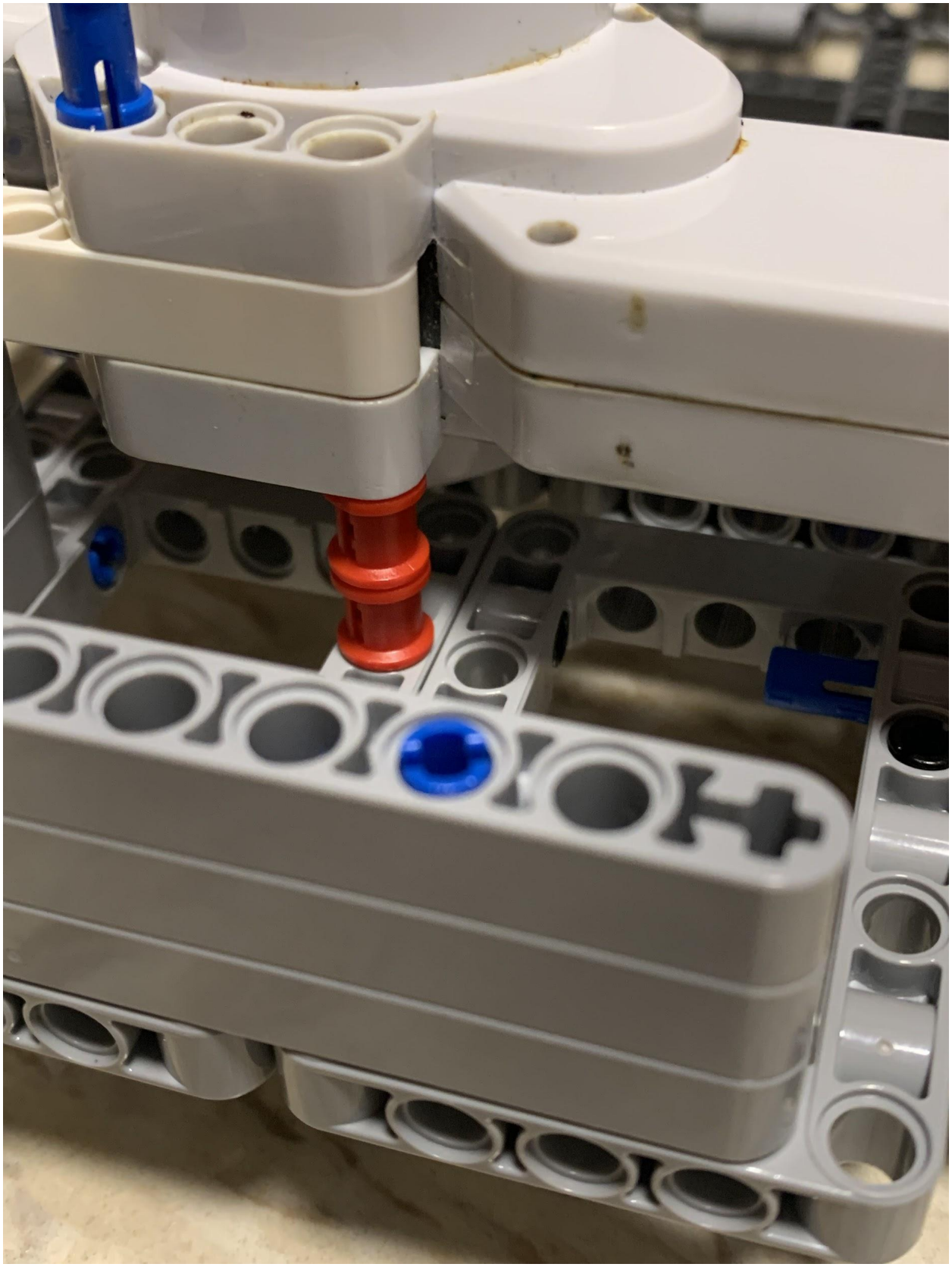
```
#!/usr/bin/env python
#
# GrovePi Example for using the Grove Line Finder
(http://www.seeedstudio.com/wiki/Twig_-_Line_Finder)
#
# The GrovePi connects the Raspberry Pi and Grove sensors.  You can learn more about GrovePi
here:  http://www.dexterindustries.com/GrovePi
#
# Have a question about this example?  Ask on the forums here:
http://forum.dexterindustries.com/c/grovepi
#
'''
## License

The MIT License (MIT)

GrovePi for the Raspberry Pi: an open source platform for connecting Grove Sensors to the
Raspberry Pi.
Copyright (C) 2017  Dexter Industries

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
```

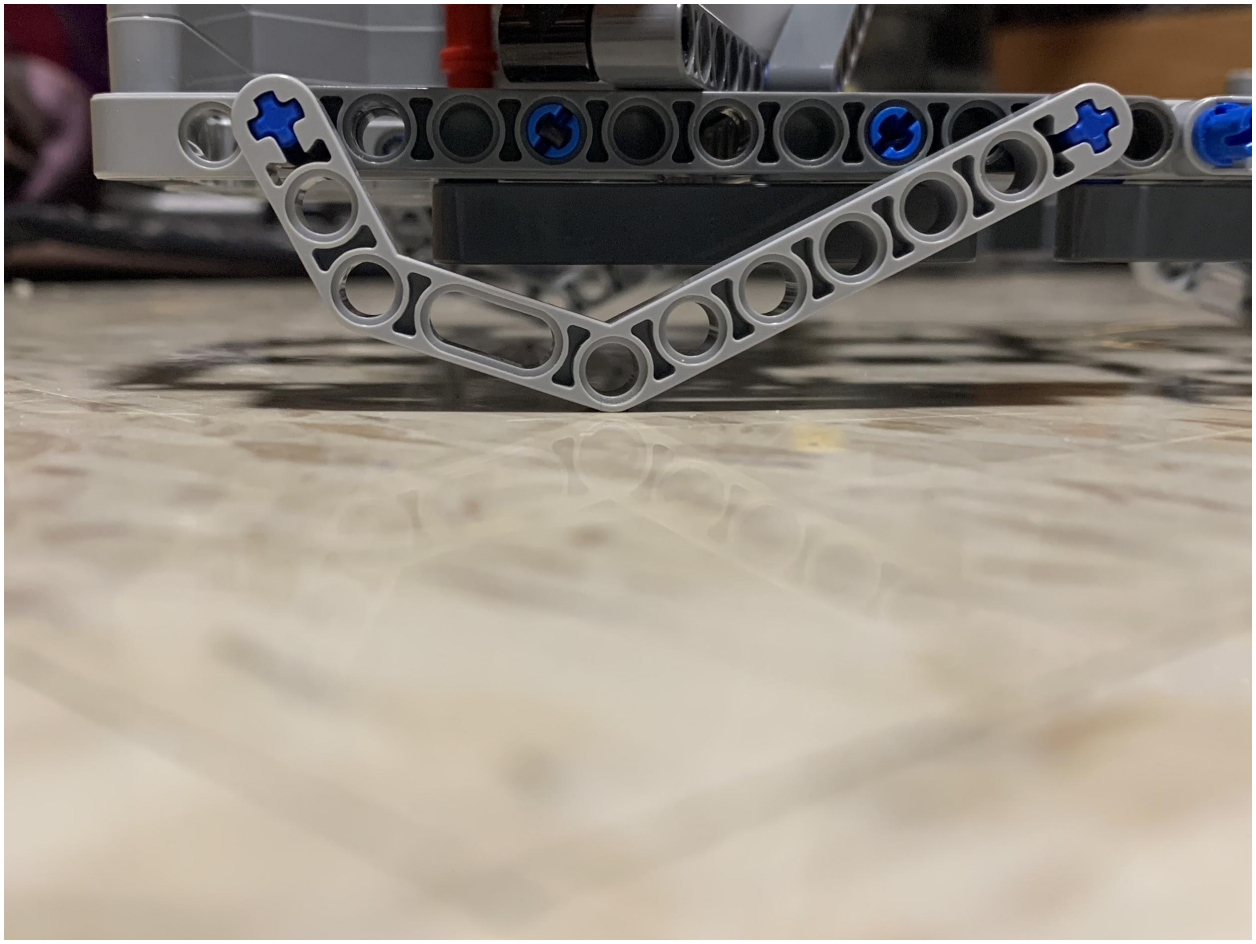
```
The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT
SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN
THE SOFTWARE.
'''

from __future__ import print_function # use python 3 syntax but make it compatible with python
2
from __future__ import division

import time
import grovepi
import brickpi3

BP = brickpi3.BrickPi3() # Create an instance of the BrickPi3 class. BP will be the BrickPi3
object.

# Connect the Grove Line Finder to digital port D7
# SIG,NC,VCC,GND

hall = 4
line_finder_l = 7
line_finder_r = 8

grovepi.pinMode(line_finder_r,"INPUT")
grovepi.pinMode(line_finder_l,"INPUT")
grovepi.set_bus("RPI_1")
```

```
while True:
    try:
        t_end = time.time() + 10
        if (grovepi.digitalRead(line_finder_r) == 1) and (grovepi.digitalRead(line_finder_l) == 0)
or (grovepi.digitalRead(line_finder_r) == 0) and (grovepi.digitalRead(line_finder_l) == 1):
            while time.time() < t_end:
                if (grovepi.digitalRead(line_finder_r) == 1) and (grovepi.digitalRead(line_finder_l) ==
1):
                    BP.set_motor_power(BP.PORT_A, 30)#50
                    BP.set_motor_power(BP.PORT_B, 30)#50

                if (grovepi.digitalRead(line_finder_r) == 1) and (grovepi.digitalRead(line_finder_l) ==
0):
                    BP.set_motor_power(BP.PORT_A , -10)#-30
                    BP.set_motor_power(BP.PORT_B, -10)#-30
                    time.sleep(.3)
                    BP.set_motor_power(BP.PORT_A, 50)#50,40
                    BP.set_motor_power(BP.PORT_B, -20)#-20
                    time.sleep(.5)

                if (grovepi.digitalRead(line_finder_r) == 0) and (grovepi.digitalRead(line_finder_l) ==
1):
                    BP.set_motor_power(BP.PORT_A , -10)#-30
                    BP.set_motor_power(BP.PORT_B, -10)#-30
                    time.sleep(.3)
                    BP.set_motor_power(BP.PORT_A, -20)#-20
                    BP.set_motor_power(BP.PORT_B, 50)#50,40
                    time.sleep(.5)
                if (grovepi.digitalRead(line_finder_r) == 0) and (grovepi.digitalRead(line_finder_l) ==
0):
                    BP.set_motor_power(BP.PORT_A, 30)#50
                    BP.set_motor_power(BP.PORT_B, 30)#50
                    #time.sleep(.1)
                if (grovepi.digitalRead(hall) == 1):
            #print(grovepi.analogRead(hall))
                    BP.set_motor_power(BP.PORT_A, 0)
                    BP.set_motor_power(BP.PORT_B, 0)
                    time.sleep(3)
```

```
        BP.set_motor_power(BP.PORT_D, 50)
        time.sleep(.35)
        BP.set_motor_power(BP.PORT_D, 0)
        time.sleep(600)



# Return HIGH when black line is detected, and LOW when white line is detected
if (grovepi.digitalRead(line_finder_r) == 1) and (grovepi.digitalRead(line_finder_l) == 1):
    BP.set_motor_power(BP.PORT_A, 45)#50
    BP.set_motor_power(BP.PORT_B, 45)#50
    print("stop")
if (grovepi.digitalRead(line_finder_r) == 1) and (grovepi.digitalRead(line_finder_l) == 0):
    BP.set_motor_power(BP.PORT_A , -10)#-30
    BP.set_motor_power(BP.PORT_B, -10)#-30
    time.sleep(.3)
    BP.set_motor_power(BP.PORT_A, 40)#50,40
    BP.set_motor_power(BP.PORT_B, -20)#-20
    time.sleep(.5)
    print("right detected black")
else:
    print("right detected white")
if (grovepi.digitalRead(line_finder_r) == 0) and (grovepi.digitalRead(line_finder_l) == 1):
    BP.set_motor_power(BP.PORT_A , -10)#-30
    BP.set_motor_power(BP.PORT_B, -10)#-30
    time.sleep(.3)
    BP.set_motor_power(BP.PORT_A, -20)#-20
    BP.set_motor_power(BP.PORT_B, 40)#50,40
    time.sleep(.5)
    print("left detected black")
else:
    print("left detected white")
if (grovepi.digitalRead(line_finder_r) == 0) and (grovepi.digitalRead(line_finder_l) == 0):
    BP.set_motor_power(BP.PORT_A, 45)#50
    BP.set_motor_power(BP.PORT_B, 45)#50
    print("test")
print(grovepi.digitalRead(hall))
if (grovepi.digitalRead(hall) == 1):
    #print(grovepi.analogRead(hall))
    BP.set_motor_power(BP.PORT_A, 0)
```

```
        BP.set_motor_power(BP.PORT_B, 0)
        time.sleep(3)
        BP.set_motor_power(BP.PORT_D, 50)
        time.sleep(.35)
        BP.set_motor_power(BP.PORT_D, 0)
        time.sleep(600)

# when stopping turn on motor for the back


    time.sleep(.1)

except KeyboardInterrupt:
    BP.set_motor_power(BP.PORT_A, 0)
    BP.set_motor_power(BP.PORT_B, 0)
```





Meeting 12: 12/2/21 (2:30-4:00)
Electronic Signatures: Ella Barnes, Heather Mello, John Kang, Fahim Hossain

- Planning for in-class workday (12/3/21)
    - Test sensors to see which ones work best together

Meeting 13: 12/3/21(11:30-1:20)
Electronic Signatures: Ella Barnes, Heather Mello, John Kang, Fahim Hossain

- Testing sensors
- Drafting report and presentation
- Assign roles for report
    - Cover letter: Ella
    - Executive Summary: Heather
    - Design Considerations: Ella
    - Macro Physical Analysis: Full group
    - Scaling to official mars project: Ella
    - Results and Discussion: Heather
    - Conclusions and Recommendations: Heather
- Assign roles for presentation
    - Roughly follows report roles
    - Introduction and conclusion to be assigned later

Meeting 14: 12/4/21 (3:15- 4:20)
Electronic Signatures: Ella Barnes, Heather Mello, John Kang, Fahim Hossain

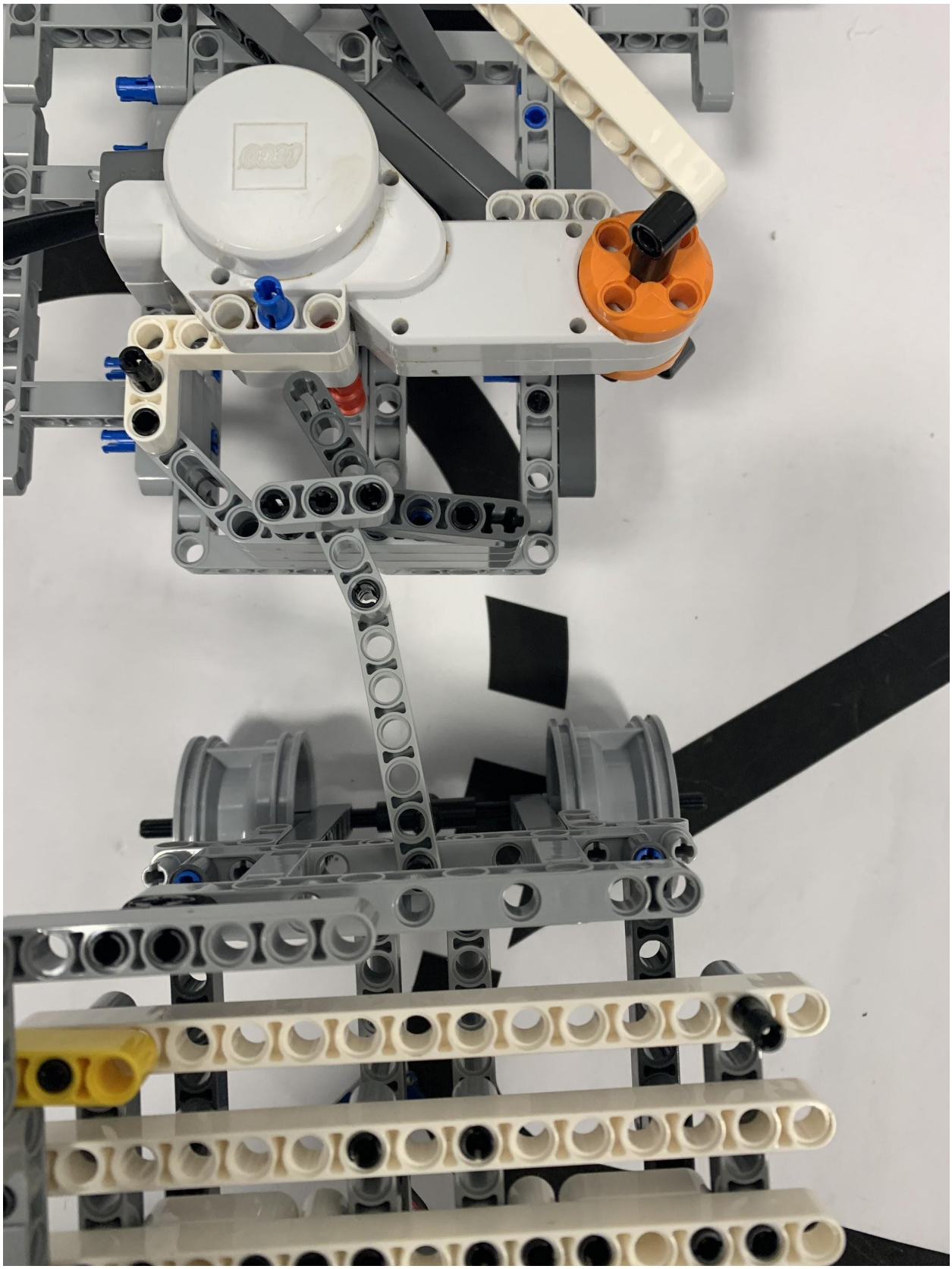
- Worked on report
- General testing

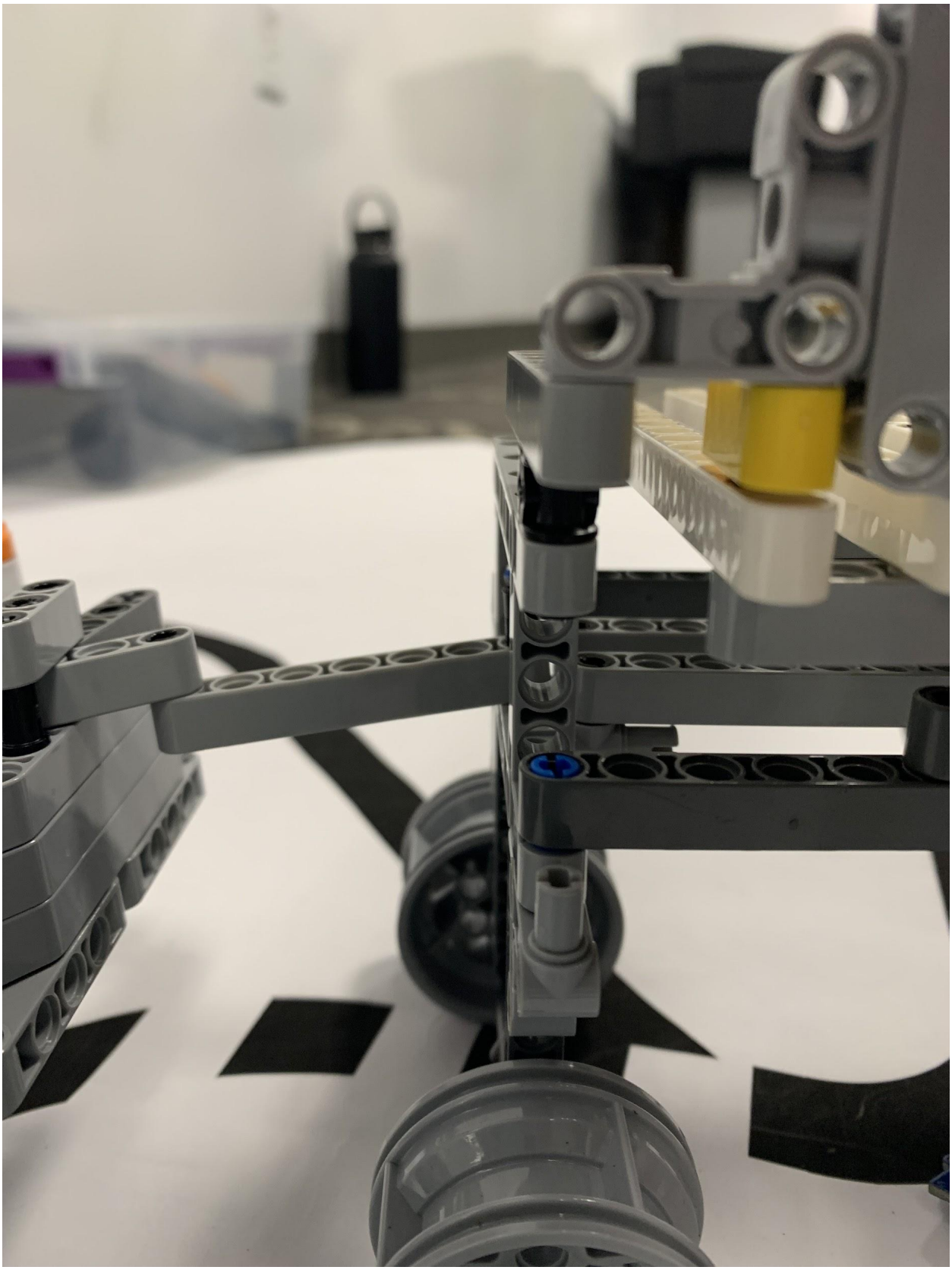
Meeting 15: 12/5/21 (2:00-
Electronic Signatures: Ella Barnes, Heather Mello, John Kang, Fahim Hossain

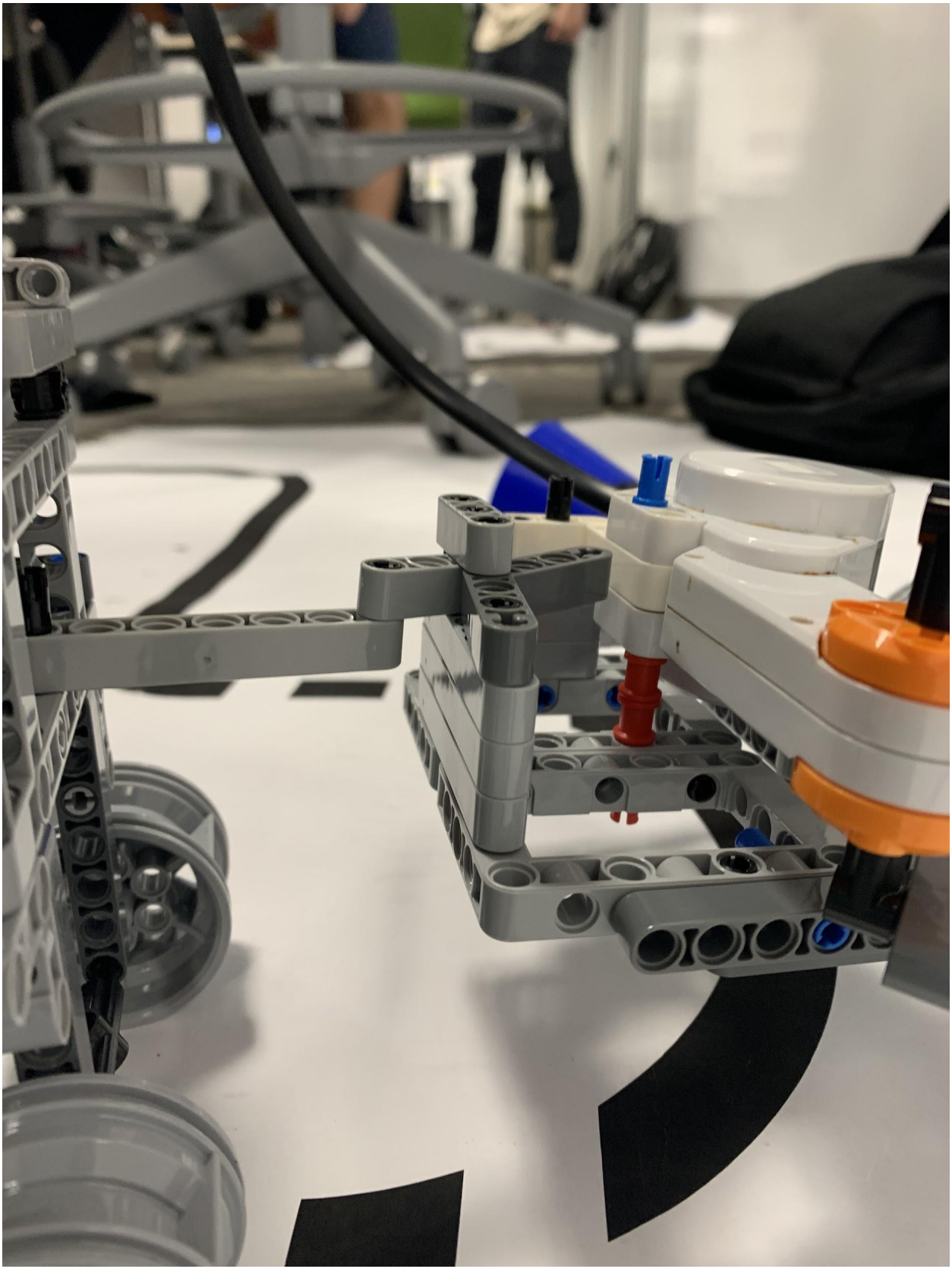
- General testing in preparation for final demonstration
- Worked on drafting the writeup
- Modified trailer attachment--attached at one point to allow for better turning and added stability to make up for missing attachment point
- Updated the engineering specifications table from DSR:

Customer Need	**Technical Need**	**Technical Requirement**	**Target Value**
Does not drop cargo inappropriately	Maximum Acceleration		
Powerful motor	Maximum Axial Torque Output	20N*cm or greater	20 N*cm
Is able to make different size turns when necessary	Turning Radius	1.5 inches to 2.5 inches (3.81 cm to 6.35 cm)	2.0 inches (5.08 cm)
Transporting cargo from location to location without dropping or tipping	Max cargo dimensions (Area of base)	45.01 cm^2 to 127.68 cm^2	127.68 cm^2
Speed restriction	Maximum speed	15 cm/s to 30 cm/s	22.5 cm/s

EV3 large servo motor 45502: MINDSTORMS®: Buy online AT THE OFFICIAL LEGO® shop us. 45502 | MINDSTORMS® | Buy online at the Official LEGO® Shop US. (2017, March 9). Retrieved December 5, 2021, from https://www.lego.com/en-us/product/ev3-large-servo-motor-45502.







Meeting 16: 12/6/21 (11:30-1:20)
Electronic Signatures: Heather Mello, Ella Barnes, John Kang, Fahim Hossain

- Worked on presentation
- Testing

Meeting 17: 12/7/21 (4:30-6:00)
Electronic Signatures: Heather Mello, Ella Barnes, Fahim Hossain, John Kang

- Finalized presentation for tomorrow
- Completed testing for physical analysis section of report
- Assigned roles for presentation
    - Heather: Abstract and project management (slides 2-6)
    - Ella: Hardware design philosophy (slides 7-10)
    - Fahim: Software design philosophy (slides 11-13)
    - John: Positive/negative attributes, conclusion, and areas for improvement (slides 14-16)
- Main body cross section: 5 ½ in x 7 ½ in (for drag calculation)
- Time to go 1m: 3.51 s with no cargo, 3.95 s with cylinder cargo
- Trailer dimensions (Max cargo): 4 ½ in x 6 in
- Acceleration:
    - .16 m/s^2 without cargo
    - .13 m/s^2 with cargo

Meeting 18: 12/10/21 (3:30-6:00)
Electronic Signatures: Heather Mello, Ella Barnes, John Kang, Fahim Hossain

- Finished written report
- Appended decision matrices made earlier in the design process (forgot to include in notebook on day of creation)

		Alternatives					
		Weight	4 Wheels, front wheel drive	4 Wheel Drive	Treads	Motorcycle Wheels	Medium Size Wheels
Criteria	Height	2	2	2	0	2	1
	Width (want it to be narrow)	2	1	1	1	1	0
	Ease of building	1	1	1	0	2	2
	Structural Stability	3	2	2	1	1	2
	Efficient Use of Materials	3	2	0	1	1	1
	Total Score:		19	13	8	14	13
	Ranking Scale:						
	A score of 2 indicates yes						
	A score of 1 indicates partial yes						
	A score of 0 indicates no						

			Alternatives		
		Weight	Garbage Truck	Floor slides out from underneath	Trailer with arm
Criteria	Ease of building	1	1	1	2
	Structural Stability	2	2	1	2
	Efficient Use of Materials	2	2	2	2
	Total Score:		9	7	10
	Ranking Scale:				
	A score of 2 indicates yes				
	A score of 1 indicates partial yes				
	A score of 0 indicates no				