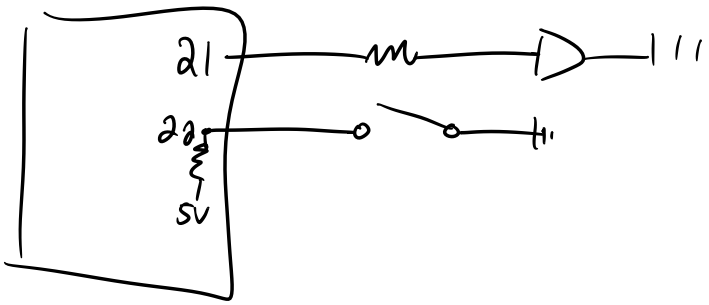
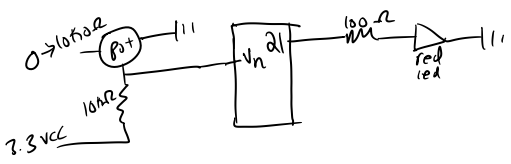
Preliminaries:

Digital Input/Output `(digitalRead(); digitalWrite();)`
4.2.1 Use ESP32 GPIO pins, a switch, an LED and resistors to read the state of a switch and light an off board LED when the button is pressed. **Submit your code.**

Submitted.



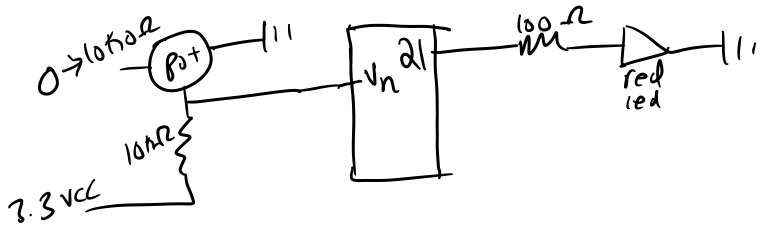
Analog Input/Output `(analogRead(); ledcWrite();)`
4.2.2 Add a potentiometer in a voltage divider circuit to be read as an analog signal and control the 50% duty cycle blinking speed of an LED using the ledc commands based on the reading. The LED should switch from off to 100% duty cycle depending on pot position, at a frequency you choose ranging from 0.1Hz to 10Hz. **If you don't do the next step, then submit your code that uses ledc for this, a drawing of your circuit, and a video link showing the LED change brightness as you turn the pot.**



Link : https://drive.google.com/file/d/1gNancwckn4xREkmrnqiKBtlLgyVMUrnw/view?usp = sharing

WiFi (TCP)

4.2.3 Building on 4.2.2, add code that will create a website that will display the frequency and duty cycle using your individually assigned IP address. **Submit your code for this, a drawing of your circuit, and a video link showing your LED as you turn the pot and the webpage displaying at the same time.**



Link: https://drive.google.com/file/d/1gVmz-n8fFcieP9Wq_FWUjyYJyXQZhrpt/view?usp = sharing

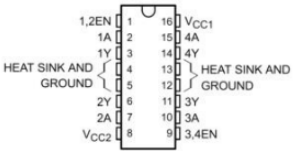
PWM and motor control (LEDC)

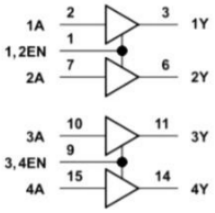
4.2.4 Generate PWM from the ESP32 using the LEDC timer functions and control the speed and direction of a motor (through a motor driver from 4.1) with a potentiometer (from 4.2.2). **Submit code on canvas. Demonstrate the motor changing speed and direction, with variable speeds in both directions. Show a TA for check off. Or if you prefer, you may submit a video, but in the video show your face before demonstrating.**

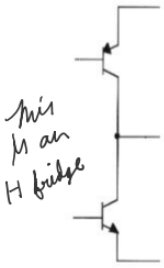
SN754410 Quad Half-H Driver

6 Pin Configuration and Functions

NE PACKAGE
(TOP VIEW)



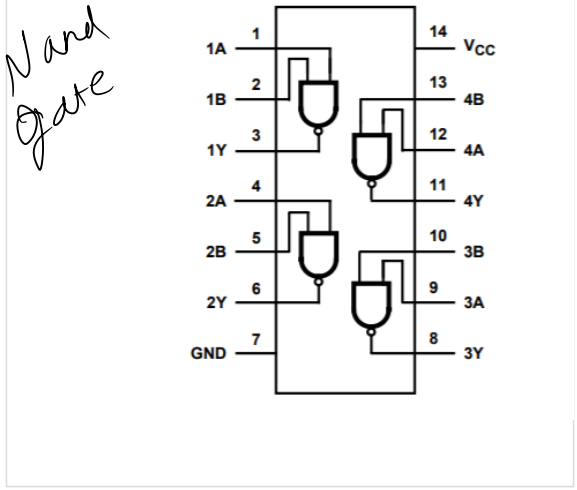




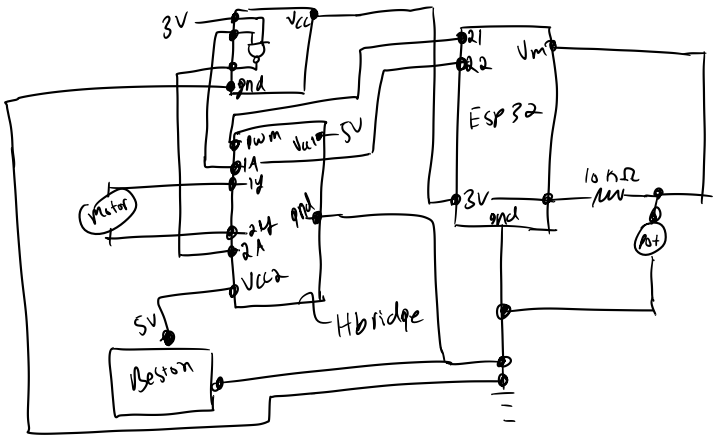
Pin Functions

PIN		TYPE	DESCRIPTION
NAME	NO.		
1,2EN	1	I	Enable driver channels 1 and 2 (active high input)
<1:4>A	2, 7, 10, 15	I	Driver inputs, non-inverting
<1:4>Y	3, 6, 11, 14	O	Driver outputs
GROUND	4, 5, 12, 13	—	Device ground and heat sink pin. Connect to circuit board ground plane with multiple solid vias
V_{CC2}	8	—	Power VCC for drivers 4.5V to 36V
3,4EN	9	I	Enable driver channels 3 and 4 (active high input)
V_{CC1}	16	—	5V supply for internal logic translation

https://www.ti.com/lit/ds/symlink/sn754410.pdf



• wires are where wires are connected



Link: https://drive.google.com/file/d/1gjvwT6EZh1yNq_MSx3fY0JbcRQZ50Bf0/view?usp = sharing

Wifi Through the internet

4.2.5b Bring your mobile base to your TA/Coach in the GMlab. Arrange for a time when they are at the GM lab for you to test-drive your mobile base from home through the internet. Show that you can command your mobile base to go straight and turn when you want. **Get a check off from a TA of you driving your mobile base from home through the internet while the mobile base is in the GM Lab. Submit code on Canvas.**

Checked off

Wifi at home

4.2.5a Drive the mobile base you create at home using your home WiFi. (Please notify the staff if you don't have home WiFi). You may use the supplied MEAM510 javascript code, or create your own UDP or TCP web-based interface. Note that for the next parts HTML (TCP) is the only option through the internet, so it may be more useful to use TCP. **Get a check off from a TA or submit a video of you driving your mobile base in your home via wifi. Submit code on Canvas if different from the next step (otherwise indicate that it is the same).**

Link: https://drive.google.com/file/d/1glK9tZuVAFjo2cN6VJbWSAbfTl-m5HLu/view?usp = sharing

Mobile Base Race

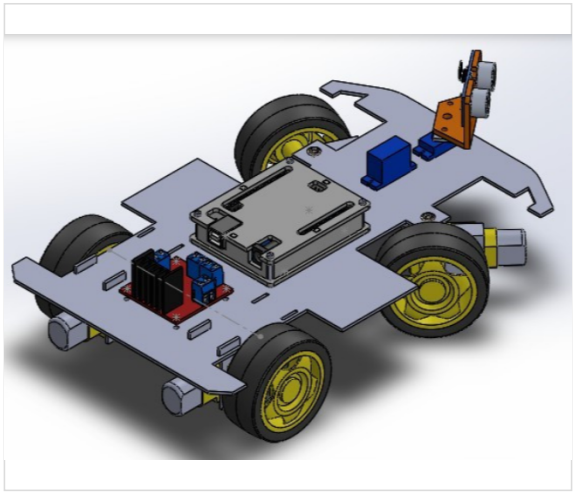
4.2.6a Arrange for a time when you and your TA/Coach's other teams can drive at the same time. Race 3 times around the center object in the GM lab. Between activities 4.2.5 and 4.2.6 you may want to tweak your mobile base for better traction on the wood panel, for more controllability and/or for speed. **Submission will be your time performance in the race compared to the whole class, which will count as 2% of this lab grade.**

Checked off and completed race. Time was 656 seconds I kept getting connection drop offs.

Final Design

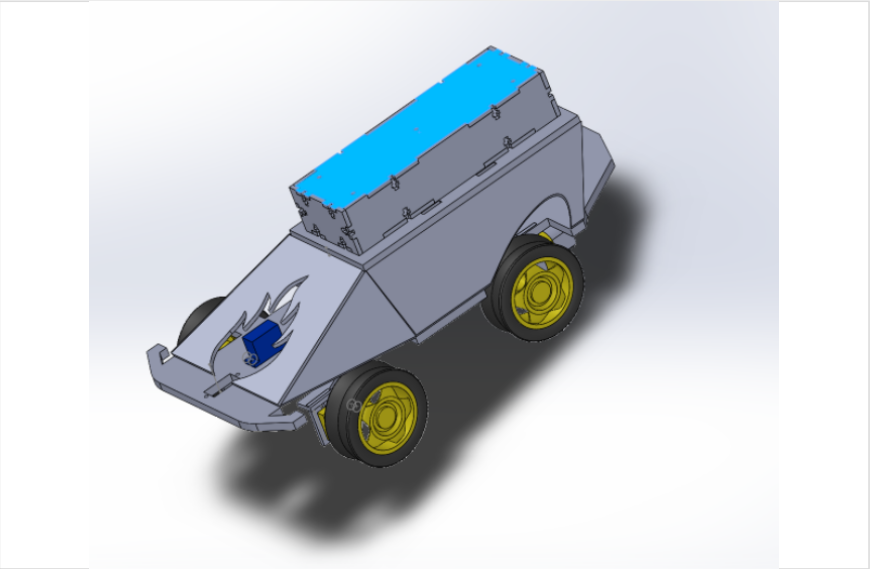
1) I used ackerman steering to control the direction of my mobile base. I got the start of the design from grab cad but I modified it heavily to have a stronger more robust and asthetically pleasing design. I added a battery pack, wire pack, and shell design along with size and part adjustments to get the pieces to be laser cutable.

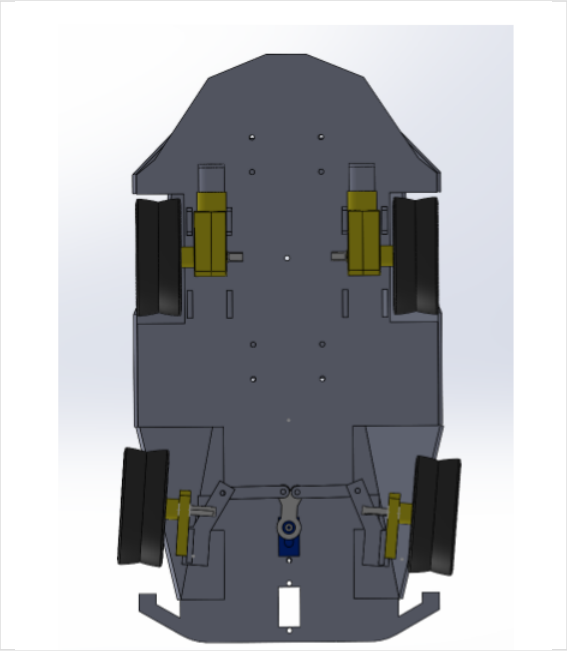
Borrowed design:

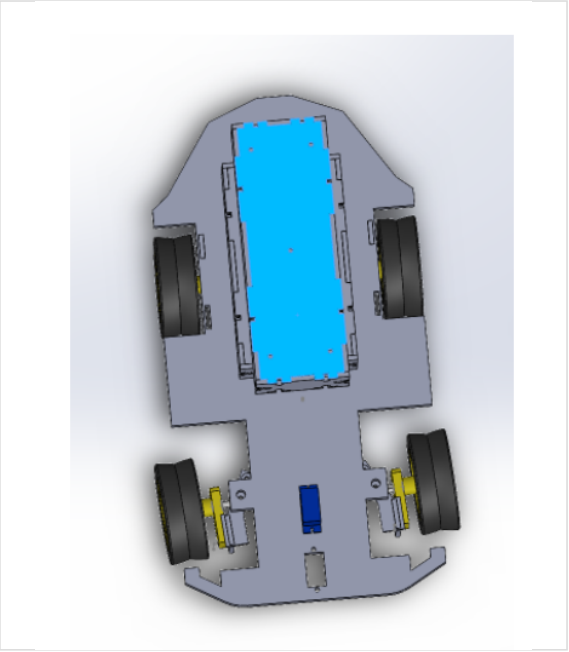


I consulted with my TA Tony about whether using these cad files were academically honest, and he said it was fair.

My design:

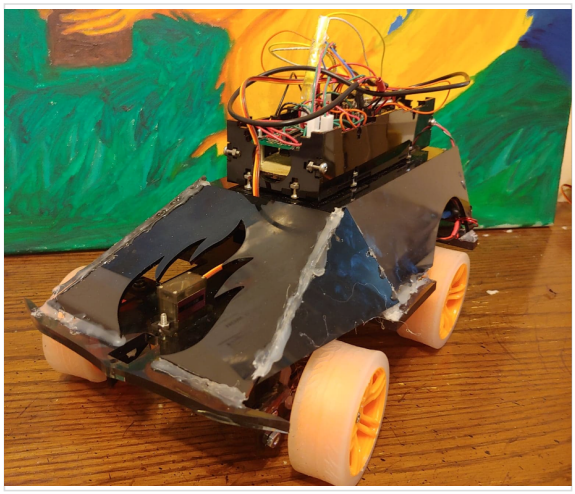


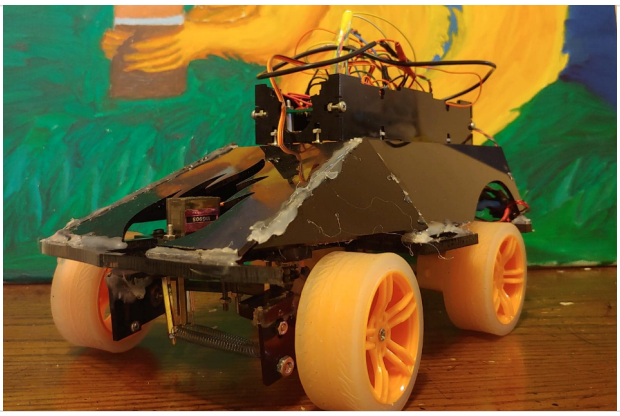




I used the a modified joystick web page to control this robot. I also had to make various physical

modifications to get things to work. I had to add in suspension to prevent front wheel sag. Figuring this part out took a lot of effort and work, but I eventually got the front wheels to turn and not grind against the shell.





2) There are various improvement I would make.
    1) I would have made the circuit box larger to accomodate the extra wires.

2) I would redesign the steering system to be more reliable. The current suspension system is very "janky" . I would design this to be more robust.
3) I would use arcylic cement instead of hot glue to keep the shell together.
4) I would make the L bracket on the wheel thicker to be prevent cracking. I could also use acrylic cement here to make it more robust.
5) I would have soldered the nand chip into the board to prevent it from falling. After the races I tried this and it didn't damage the chip and was far more reliable. There were several times where the nand chip got dislodged from the socket pins
6) I would not use the joystick controller because it makes it hard to move forward in a straight line. I would use the tank controller.

3) See the other file for code submission

4) **BOM:**

Part Name	Price	Quantity
Wheels	6	4
Double A Batteries and holders	10.48	12
uxcell 693ZZ Deep Groove Ball Bearings Z2	9.99	10
Acrylic	12	3
M3 screw kit	10	
Hot glue	5	
Perf boards for m3 standoffs 3x2 inches	5	2
Exp32	15	
4Pcs MG90S 9g Servo Motor Micro Metal Gear for Robot Car Plane RC Helicopter Arduino	14.88	1
m3 standoffs 3 cm height	3	4
m3 standoffs 2 cm height	3	10
Total Price	94.35	

5) circuit diagram