



Floor plan EE632

EECN30169/535317
Robot Hockey Arena

TA

1

2

3

4

5

6

7	8	9	10
---	---	---	----

11

Checkpoint #2

Demo Due : 10/11/2024

Report Due : 10/18/2024

Outline

- CP2 Supplies
- Motor with Encoder
- L298N
- Motor control system
- Checkpoint #2 Assignment

CP #2 Supplies

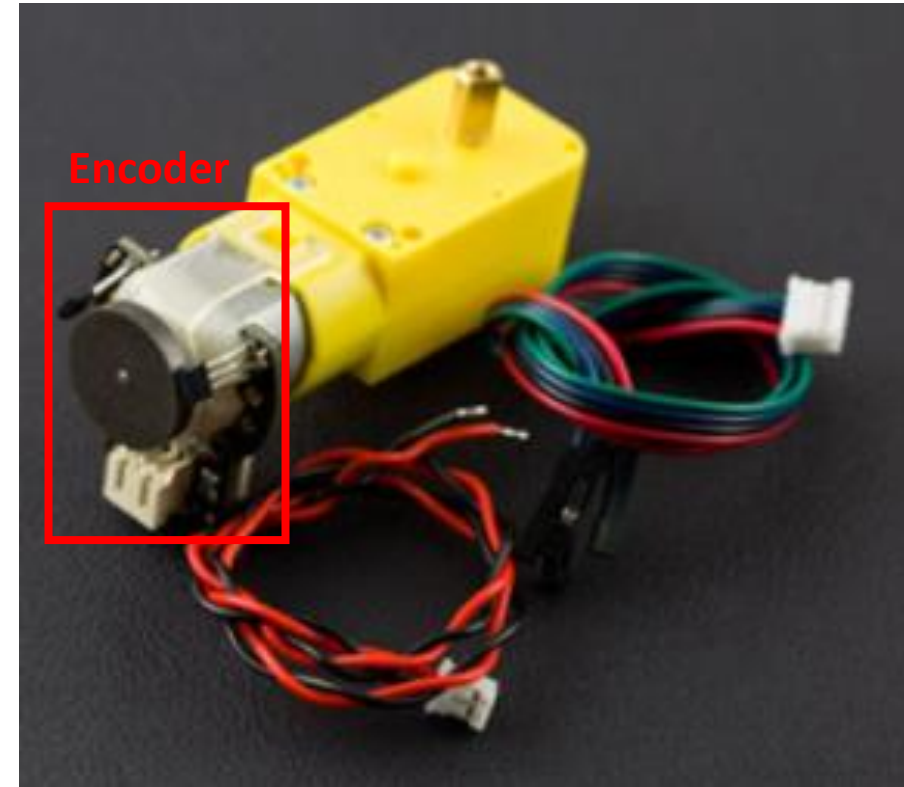


Checkpoint#2 Material List			
1	Chassis	5	L298N Motor driver module
2	DC Motor x 2	6	Li-po battery
3	Wheel x 2	7	A pack of screws
4	Caster wheel	8	Screwdriver
Team _____			

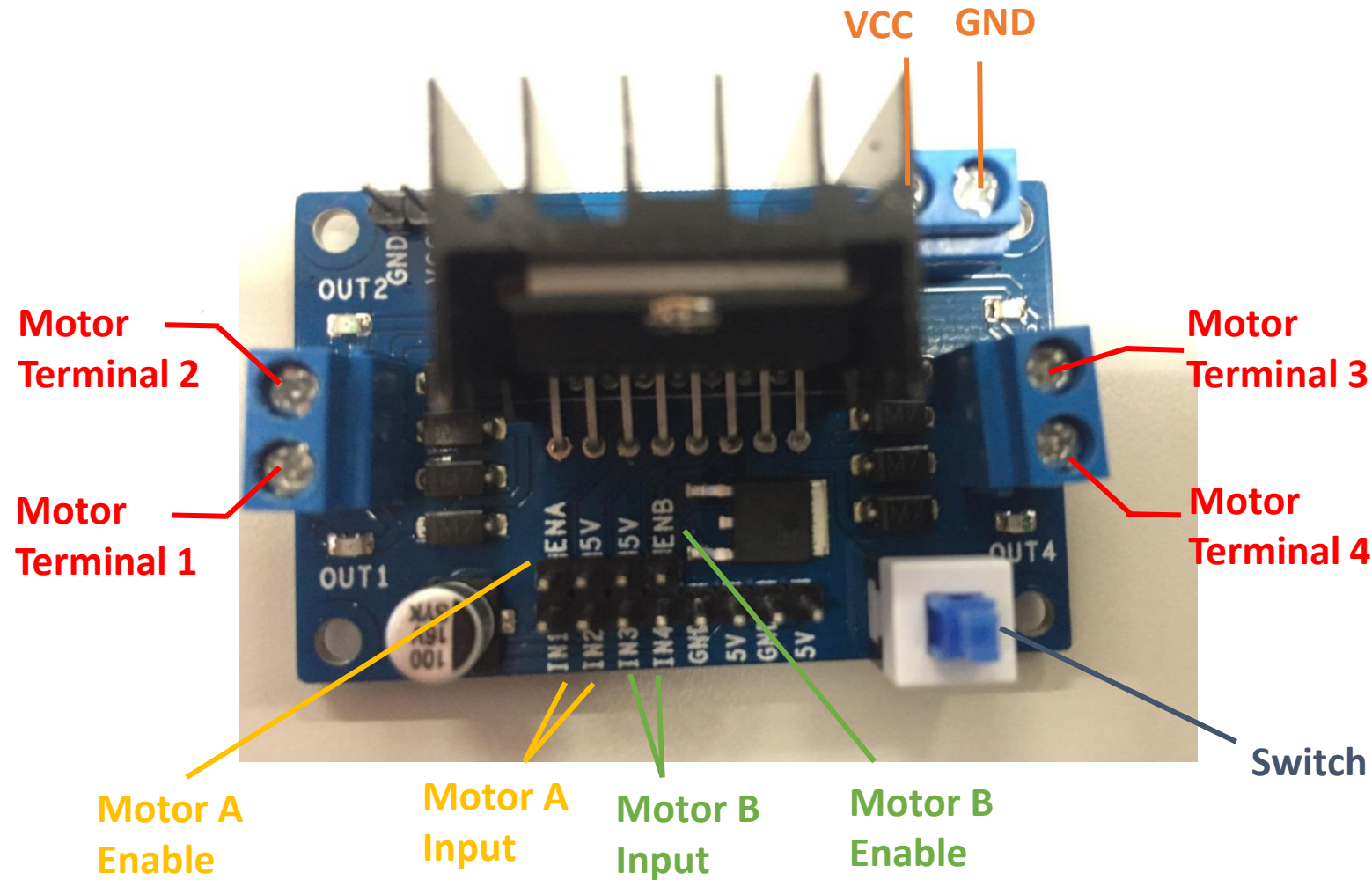


Motor with Encoder

- Two gear head motors are used for the robot. The motor is with an encoder.
- The motor is with a **120:1** gearbox and an integrated quadrature encoder, which provides **16** pulse counts per revolution.
- So, it gives **1920** pulse counts for one turn of the wheel shaft.
 - $120 \times 16 = 1920$

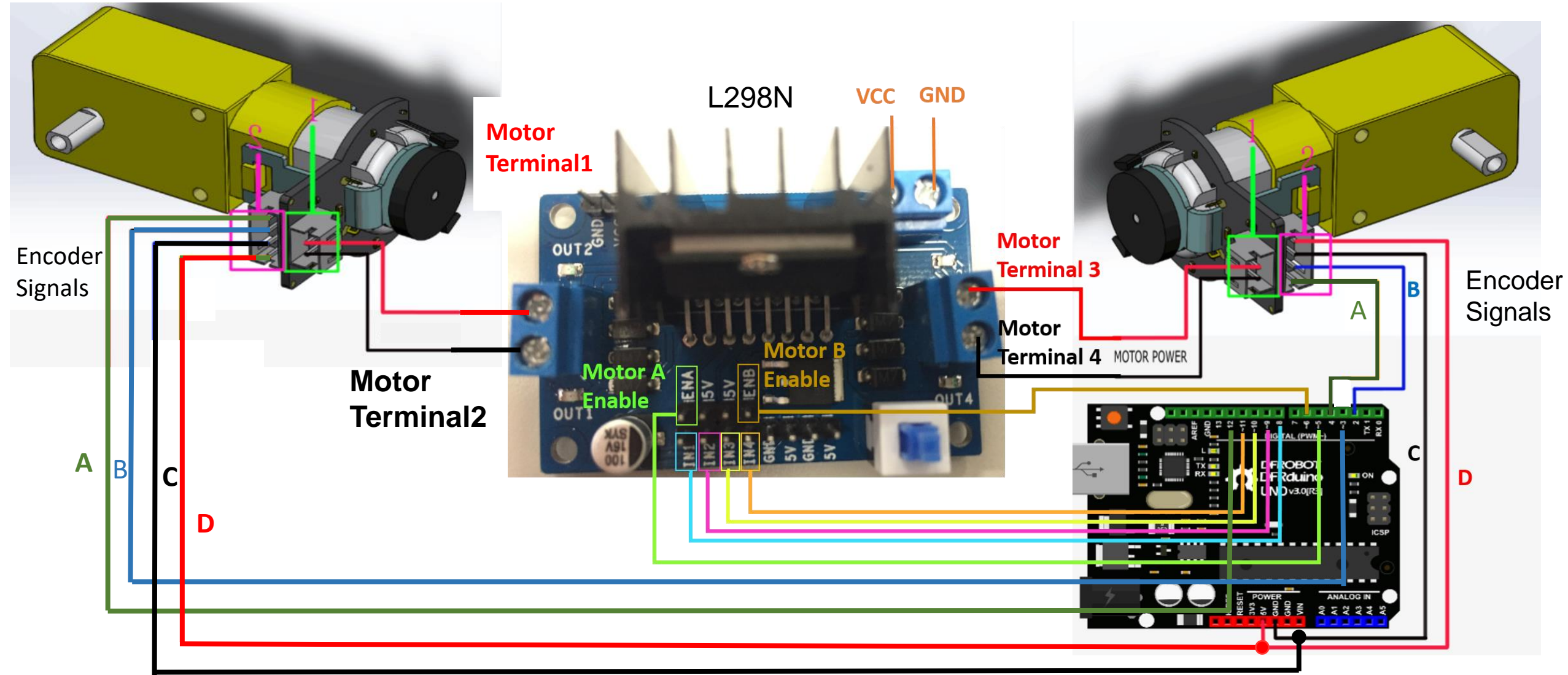


L298N H-Bridge



- Double H-bridge driver module
- Input voltage is given around 7V to 12V, can supply 5V for motors
- IN1, IN2, IN3 and IN4 : High/Low pulse for rotation direction
- ENA, ENB: PWM for speed control

Motor control system



Connections for L298N and Encoders

L298N

L298n	Arduino	Description
ENA	5	Motor A PWM
ENB	6	Motor B PWM
IN1	8	Motor A direction
IN2	9	
IN3	10	Motor B direction
IN4	11	

Encoder

Pin	Description
A	Encoder A phase output
B	Encoder B phase output
C	Encoder supply GND
D	Encoder power supply 4.5-7.5V

Motor with Encoder

- Interrupts are useful for making things happen automatically in microcontroller programs and can help solve timing problems.
- using an interrupt can free the microcontroller to get some other work done while not missing the input.
- Interrupt Port with Different Board

	Board	Int.0	Int.1	Int.2	Int.3	Int.4	Int.5
attachInterrupt()	Uno ,Ethernet	2	3				
	Mega2560	2	3	21	20	19	18
	Leonardo	3	2	0	1	7	

Control Programming

```
analogWrite(6, enA);  
digitalWrite(10, in1);  
digitalWrite(11, in2);  
analogWrite(5, enB);  
digitalWrite(8, in3);  
digitalWrite(9, in4);
```

```
void EncoderInit()  
{  
    Direction_L = true; //default -> Forward  
    Direction_R = true; //default -> Forward  
    pinMode(encoder0pinB_L, INPUT);  
    attachInterrupt(0, wheelSpeed_L, CHANGE);  
    pinMode(encoder0pinB_R, INPUT);  
    attachInterrupt(1, wheelSpeed_R, CHANGE);  
}  
  
void wheelSpeed_L()  
{  
    int Lstate_L = digitalRead(encoder0pinA_L);  
    if ((encoder0PinALast_L == LOW) && Lstate_L == HIGH)  
    {  
        int val_L = digitalRead(encoder0pinB_L);  
        if (val_L == LOW && Direction_L)  
        {  
            Direction_L = false; //Reverse  
        }  
        else if (val_L == HIGH && !Direction_L)  
        {  
            Direction_L = true; //Forward  
        }  
    }  
    encoder0PinALast_L = Lstate_L;  
  
    if (!Direction_L)  
        duration_L++;  
    else  
        duration_L--;  
}
```

Checkpoint #2 Assignment

- Purpose:
The purpose of this checkpoint is to make sure you can control the motion of DC motors by using PWM with Raspberry Pi and Arduino.
- Tasks:
Construct the basic motion platform of the robot using the chassis.
- Demonstrate your robot performing the following actions by giving PWM value to the motor individually.
 - Move forward. (25%)
 - Move backward. (25%)
 - Turn right. (15%)
 - Turn left. (15%)
 - How straight the robot can move when moving forward. (20%)

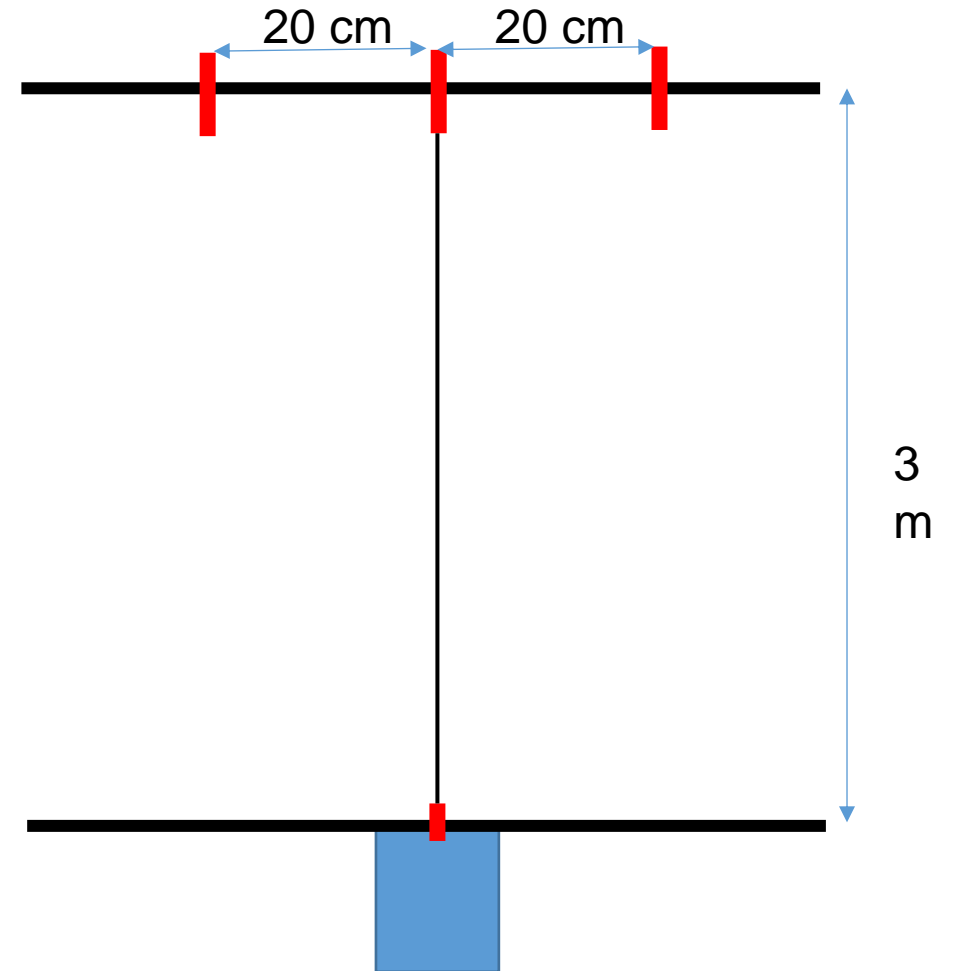
Checkpoint #2

You should send the right wheel and left wheel PWM signal command on the RPI, then send the command to Arduino to control motors in each task.

```
setting /run_id to 0fc9125a-1011-11e8-9501-b827ebaa4d9b
process[rosout-1]: started with pid [2832]
started core service [/rosout]
process[connect_arduino-2]: started with pid [2835]
process[checkpoint2-3]: started with pid [2836]
user's right is 120
user's left is 120
user's right is -100
user's left is 50
user's right is 100
user's left is 200
user's right is 0
user's left is 0
user's right is 100
user's left is 100
user's right is -50
user's left is 50
user's right is 0
user's left is 0
```

Checkpoint #2 Task 5 Scoring Rules

- Your mobile robot will move forward 3 meters until the caster wheel passes the finish line. You have two chances to challenge.
- We will measure how far the caster wheel deviate from the center point of the finish line.
- One point will be deducted for every deviation of **1 cm**.
- If the deviation is over **20 cm**, you will get 0 point in task 5.



Reference

- Micro DC Motor with Encoder
 - https://wiki.dfrobot.com/Micro_DC_Motor_with_Encoder-SJ01_SKU__FIT0450
- L298n
 - <https://kknews.cc/zh-tw/education/b5nm256.html>
- Digital Pins With Interrupts
 - <https://www.arduino.cc/reference/en/language/functions/external-interrupts/attachinterrupt/>

Deadline

- Checkpoint#2 Demo : 10/11
- Checkpoint #2 Report : 10/18