Polytechnic University of the Philippines
College of Engineering
Department of Computer Engineering
Sta. Mesa, Manila

COEN 3374 | ELECTRONICS CIRCUITS ANALYSIS AND DESIGN

FINAL PROJECT

Line Following Robot and Sumo Robot

BSCpE 3-1
Esteban, Charlene Mae DG.
Gomez, Enrico Camilo P. Jr
Guevarra, Justin Earl L.
Morita, Ami V.

March 17, 2018
Date Submitted

Engr. Mark Kervin Natividad
Professor

I. Project Description

The robot is designed to work as a Line Following Robot and Sumo Robot separately. For:

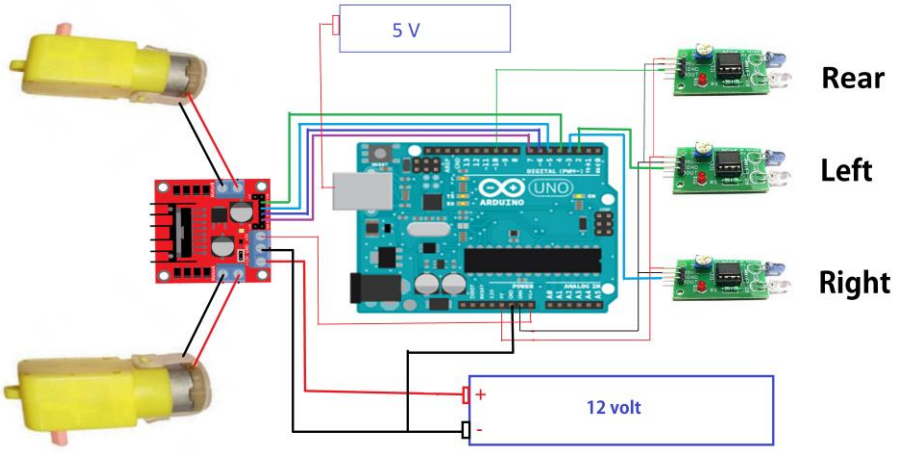
a. Line Following Robot – the Robot is programmed to finish a track. It follows the pattern of the track.
b. Sumo Robot – the Robot is programmed to prevent itself from going outside the arena. Also, it must defeat its opponent by pushing it outside the arena.

II. Definition of Terms

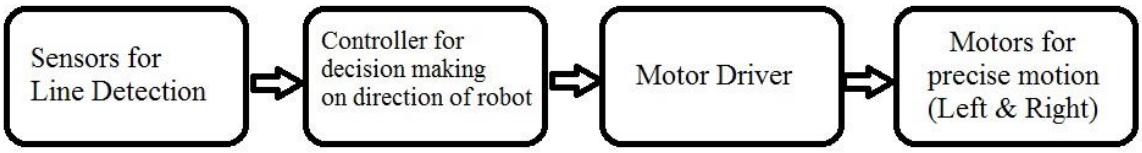
a. Line tracking sensor – a pre-defined circuit module to sense surfaces. This is used to follow or to avoid dark or light surfaces.
b. Arduino Uno – microcontroller board used to control the decision making on the direction of robot
c. Motor driver – a pre-defined circuit module to drive two motors at the same time.
d. Chassis – base frame of a robot

III. Components of Project

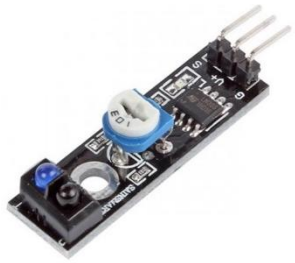
a. Schematic Diagram



b. Block Diagram

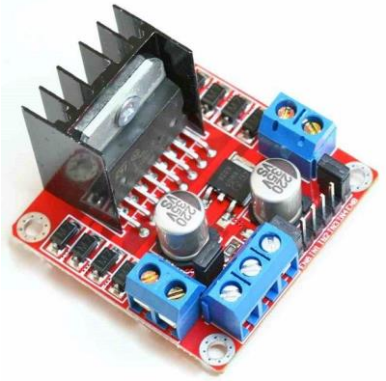


c. Peripherals
   1. Line tracking sensor – outputs low when it detects a surface other than black; outputs high when it detects black surface or nothing.



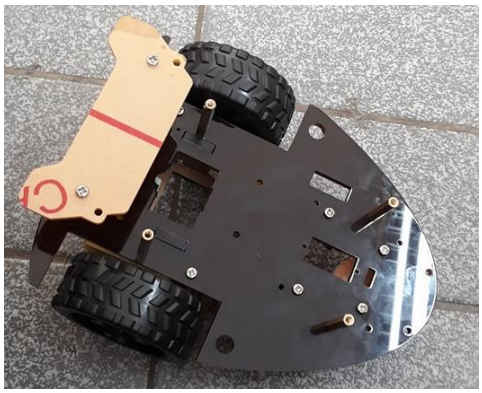
In line following robot, the two sensors at the front – left and right – are used. In sumo robot, the three sensors – left and right front sensors, and one rear sensors – are used. The left and right front sensors tell the robot when to turn left or right or move forward as it detects a black line while the rear sensor tells the robot whether there is a black line at its back.

2. Clone Arduino Uno – controls the decision making on the direction of the robot based on the ddetection of the sensors.



3. L298N Motor Driver – is a dual H-bridge motor driver which allows speed and direction control of two DC motors at the same time. The module can drive DC motors that have voltages between 5 and 35V, with a peak current up to 2A.

4. Geared DC motors with wheels – motors for precise position. HIGH output for forward motion and LOW output for backward motion.



5. Bot Chassis – for organized installation and connection of the electronic components used.

6. 8 1.5-V Batteries and Holder – for Voltage supply at L298N Motor Driver

7. Power bank and Connector for Arduino Uno – 5V supply for Arduino Uno

8. Wires – Connecting wires, Male-to-female jumper wires

IV. How the Mobot/Sumobot works?

This robot is a one-design-two-applicable-programs wherein it can be a line following robot or can be a sumo robot.

For line following robot, two front sensors are used – left and right. The robot runs on a white surface, and when it detects black line, it will avoid it. The black line should be at the center of the two front sensors. When the left sensor detects the black line, or technically detects nothing, the left motor turns backward making the robot turn left. When the right sensor detect black line, the right motor turns backward making the robot turn right. When both sensors detect white surface, the both motors turn forward making the robot run forward. When both sensors detect nothing, the both motors stop.

For the sumo robot, three sensors are used – two front sensors and one rear sensor. When two front sensors – left and right – detect white surface, both motors run forward. When both front sensors detect black surface, the robot will move backward – rightward – forward – leftward. When left front sensor detects black surface, the robot will turn left. Robot will turn right for sensing black surface on the right front sensor. Rear sensor is designed to tell the robot that there is a black surface at its back that it needs to avoid. If so, the robot will move forward – rightward – forward – leftward.

V. Procedures

1. Assemble the bot chassis. This includes installation of two geared motors with wheels.
2. Mount the Arduino Uno and L298N motor driver on the chassis.
3. Connections:
    - Positive and negative (depends on the installation on the chassis) of two motors to MotorA Pins and MotorB Pins of L298N motor driver.
    - 5V Supply to the Arduino
    - 12V Supply with switch positive to 12V Pin on the L298N motor driver and negative on GND Pin. Ground must be paralleled to the GND pin of Arduino Uno

L298N motor driver Pin	Arduino Uno Pin
IN1	7
IN2	6
IN3	5
IN4	4

    - 5V pin on the three sensors connected as parallel at 5V pin on Arduino Uno. GND pin on the three sensors connected as parallel at GND pin on Arduino. OUT pin on the three sensors:

IR Sensor	Arduino Pin
Front left	2
Right left	3
Rear	10

4. Switch on, then test the circuit. If it does not work properly, troubleshoot the program, the connections, or the electronic components.
5. Finalize the circuit connection. If running properly then design as desired.

VI. Bill of Materials

Item	Quantity	Unit Cost	Amount
Arduino Uno	1	300	300
L298N Motor Driver	1	146	146
USB Cable	1	95	95
7800 mAh Powerbank	1	300	300
TCRT5000 IR Sensor	2	124	248
Line tracking Sensor	1	150	150
Male to Female wires	15	5	75
Double A Battery	6	8	48
Ace Alkaline AA Battery	2 packs	80	160
8 Cell Battery Holder	1	32	32
Toggle switch	1 pack	24	24
Chassis	1	709	709
Galvanized Aluminum Sheet	1 – 4”x8”	240	240
Galvanized Wire	1 pack	29.75	29.75
Glue gun	1	105	105
Glue sticks	1 pack	35	35
Soldering lead	1 meter	20	20
		TOTAL EXPENSES:	**2716.75**

VII. Sources/References

https://www.youtube.com/watch?v=QL4K7n2c5WM
https://drive.google.com/drive/folders/0BwsV1jJYW9dnbzVlQjlYQ3N2QTg
https://www.electronicshub.org/arduino-line-follower-robot/

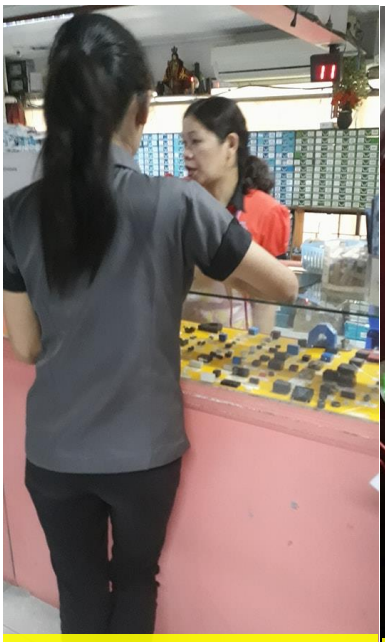
VIII. Safety and precautions in project construction

1. Make sure of correct connections. Incorrect connection may lead to destruction of electronic components.
2. Know the minimum and maximum voltage capacity of each electronic component.
3. Be careful when using soldering iron. It may lead to burning. Same with glue gun. Plug out when not in use.
4. Be careful when cutting materials like galvanized aluminum sheet for it may lead to cut wounds or worse.

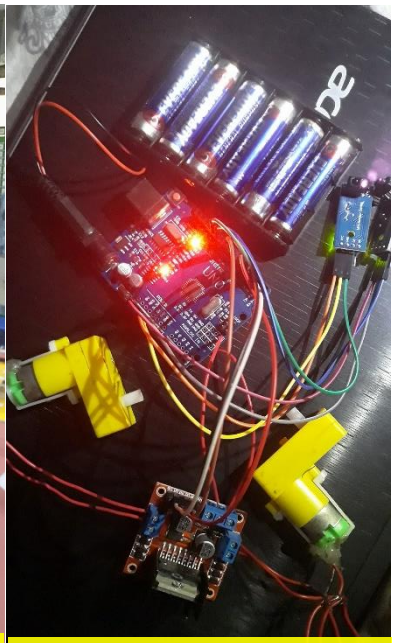
IX. Gantt Chart

	01/25/2018	02/01/2018	02/08/2018	02/15/2018	02/22/2018	03/01/2018	03/08/2018	02/15/2018	02/17/2018
Research for References									
Buying of Materials									
Circuit designing									
Actual Circuit Connections									
Coding of Line following Robot									
Troubleshooting									
Improvements									
Coding of Sumobot									
Troubleshooting									
Improvements									
Final testing without design									
Designing									
Final testing with design									

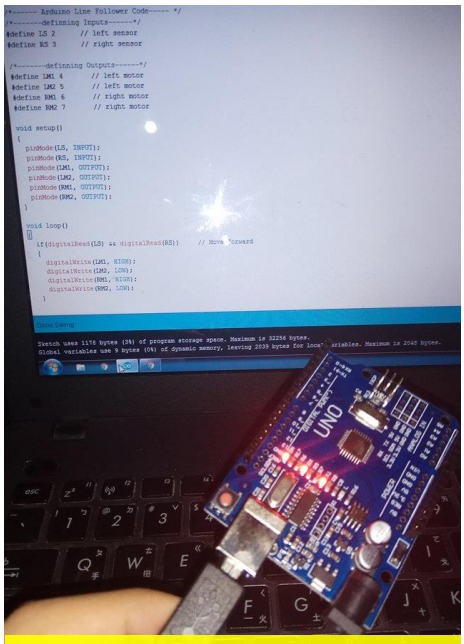
X. Project Photos



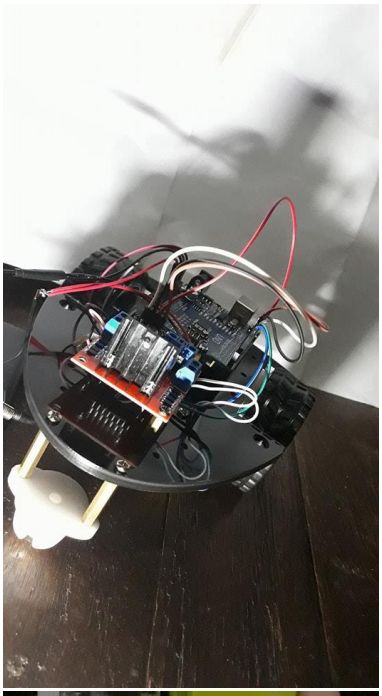
Buying of materials

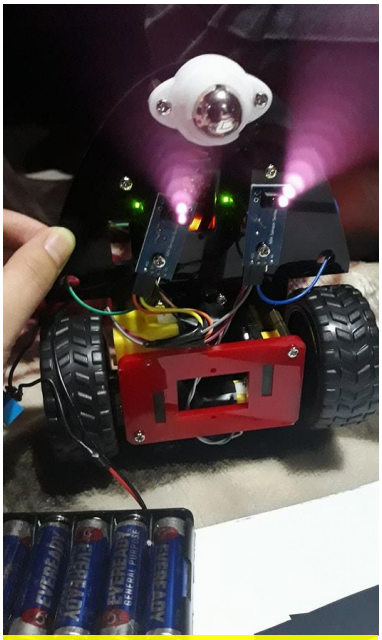


Circuit designing

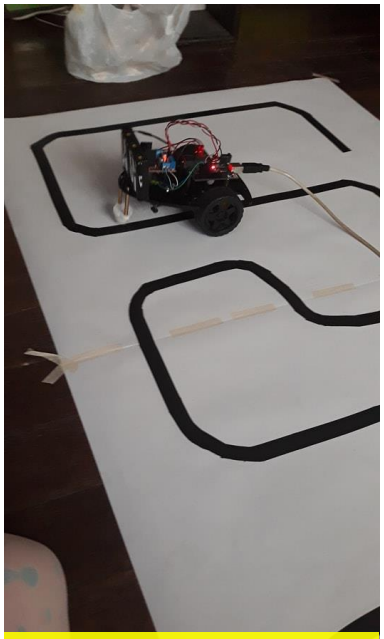


Coding

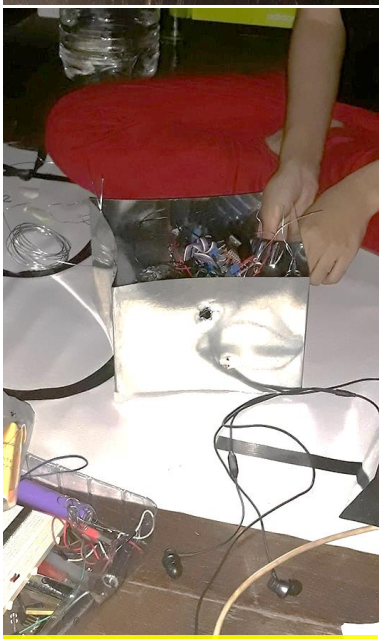




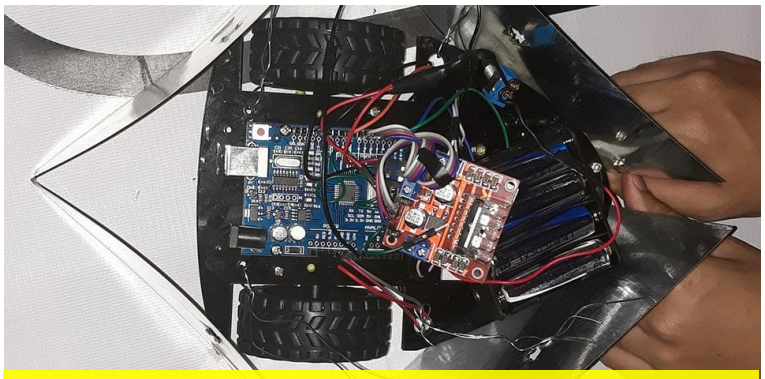
Testing/Troubleshooting



Testing



Designing



Designing

XI. Program Codes for Mobot and Sumobot

```
//LINE TRACKING ROBOT
#define LS 2 // left front sensor
#define RS 3 // right front sensor
#define LM1 5 // left motor M1a
#define LM2 4 // left motor M2b
#define RM1 7 // right motor M2a
#define RM2 6 // right motor M2b
void setup()
{
pinMode(LS, INPUT);
pinMode(RS, INPUT);
pinMode(LM1, OUTPUT);
pinMode(LM2, OUTPUT);
pinMode(RM1, OUTPUT);
pinMode(RM2, OUTPUT);
}
void loop()
{
if(!(digitalRead(LS)) &&
!(digitalRead(RS))) // Move Forward
{
digitalWrite(LM1, HIGH);
digitalWrite(LM2, LOW);
digitalWrite(RM1, HIGH);
digitalWrite(RM2, LOW);
}
if((digitalRead(LS)) &&
!(digitalRead(RS))) // turn left
{
digitalWrite(LM1, HIGH);
digitalWrite(LM2, LOW);
digitalWrite(RM1, LOW);
digitalWrite(RM2, HIGH);
}
if(!(digitalRead(LS)) &&
(digitalRead(RS))) // Turn right by
rotating right motors in forward and left
ones in backward direction
{
digitalWrite(LM1, LOW);
digitalWrite(LM2, HIGH);
digitalWrite(RM1, HIGH);
digitalWrite(RM2, LOW);
}

if((digitalRead(LS)) && (digitalRead(RS)))
// Finish line, stop both the motors
{
digitalWrite(LM1, LOW);
digitalWrite(LM2, LOW);
digitalWrite(RM1, LOW);
digitalWrite(RM2, LOW);
}
}
```

```
//SUMO ROBOT
#define LS 2 // left front sensor
#define RS 3 // right front sensor
#define BS 10 // rear sensor
#define LM1 5 // left motor M1a
#define LM2 4 // left motor M2b
#define RM1 7 // right motor M2a
#define RM2 6 // right motor M2b
void setup()
{
pinMode(LS, INPUT);
pinMode(RS, INPUT);
pinMode(BS, INPUT);
pinMode(LM1, OUTPUT);
pinMode(LM2, OUTPUT);
pinMode(RM1, OUTPUT);
pinMode(RM2, OUTPUT);
}
void loop()
{
if(!(digitalRead(LS)) &&
!(digitalRead(RS))) // Move Forward on
line
{

digitalWrite(LM1, HIGH);
digitalWrite(LM2, LOW);
digitalWrite(RM1, HIGH);
digitalWrite(RM2, LOW);
}
if((digitalRead(LS)) &&
!(digitalRead(RS))){
  digitalWrite(LM1,LOW);
  digitalWrite(LM2,HIGH);
digitalWrite(RM1,LOW);
  digitalWrite(RM2,HIGH);
  delay(500);
  loop ();
  digitalWrite(LM1, LOW);
  digitalWrite(LM2, HIGH);
  digitalWrite(RM1, HIGH);
  digitalWrite(RM2, LOW);
  delay(800);


}
if(!(digitalRead(LS)) &&
(digitalRead(RS)))//turn right
{
  digitalWrite(LM1,LOW);
  digitalWrite(LM2,HIGH);
  digitalWrite(RM1,LOW);
  digitalWrite(RM2,HIGH);
  delay(500);
  loop();
  digitalWrite(LM1, HIGH);
  digitalWrite(LM2, LOW);
  digitalWrite(RM1, LOW);
  digitalWrite(RM2, HIGH);
delay(800);
}
if((digitalRead(LS)) && (digitalRead(RS)))
// Finish line, stop both the motors
{
  digitalWrite(LM1,LOW);
  digitalWrite(LM2,HIGH);
  digitalWrite(RM1,LOW);
  digitalWrite(RM2,HIGH);
  delay(700);
```

```
loop();
  digitalWrite(LM1,LOW);
  digitalWrite(LM2,HIGH);
  digitalWrite(RM1,HIGH);
  digitalWrite(RM2,LOW);
  delay(500);
  loop();
  digitalWrite(LM1,HIGH);
  digitalWrite(LM2,LOW);
  digitalWrite(RM1,LOW);
  digitalWrite(RM2,HIGH);
  delay(500);

}
if((digitalRead(BS)))
{

  digitalWrite(LM1, HIGH);
  digitalWrite(LM2, LOW);
  digitalWrite(RM1, HIGH);
  digitalWrite(RM2, LOW);
  delay(500);
  loop();
digitalWrite(LM1,HIGH);
  digitalWrite(LM2,LOW);
  digitalWrite(RM1,LOW);
  digitalWrite(RM2,HIGH);
  delay(600);
  loop();
digitalWrite(LM1, HIGH);
  digitalWrite(LM2, LOW);
  digitalWrite(RM1, HIGH);
digitalWrite(RM2, LOW);
  delay(300);
  loop();
  digitalWrite(LM1,LOW);
  digitalWrite(LM2,HIGH);
  digitalWrite(RM1,HIGH);
  digitalWrite(RM2,LOW);
  delay(600);

}
}
```