



# MTU

Ollscoil Teicneolaíochta na Mumhan  
Munster Technological University

## Lab 6

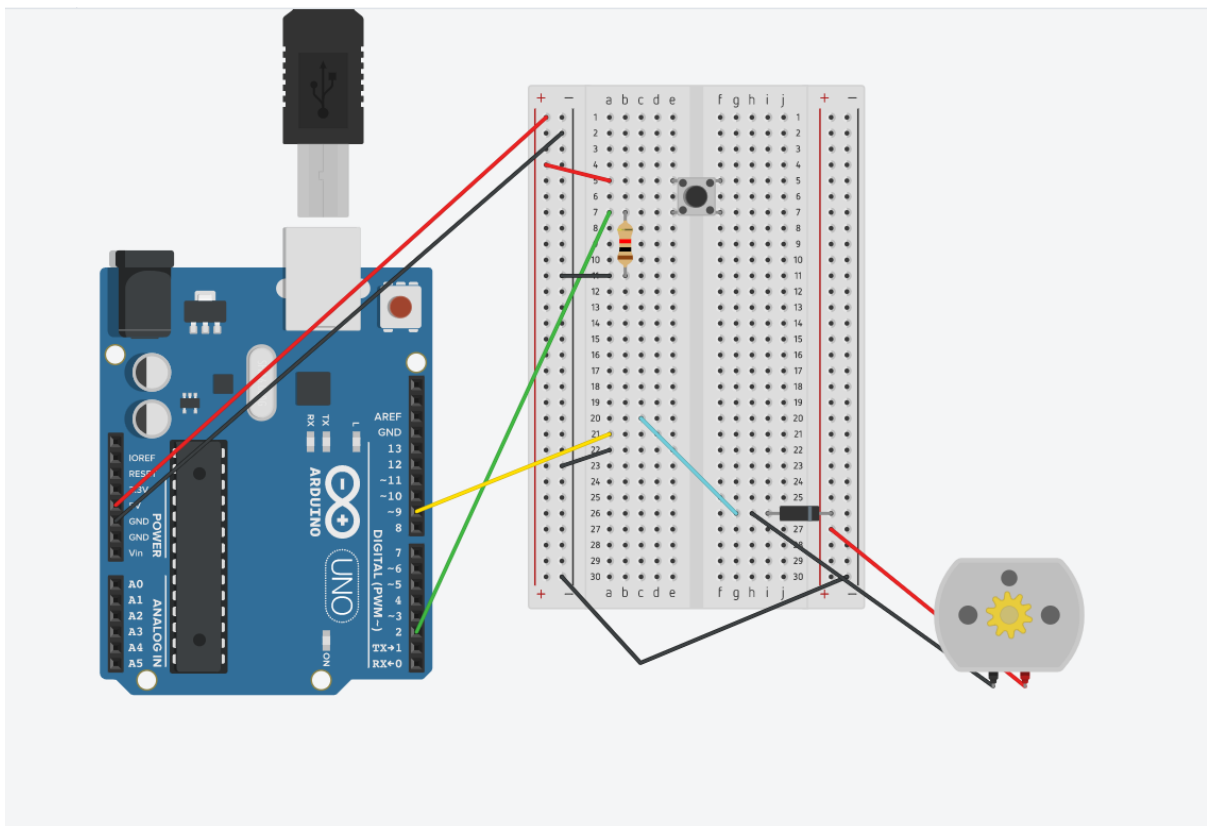
*COMP6043: Physical Computing*

Student ID: R00201303

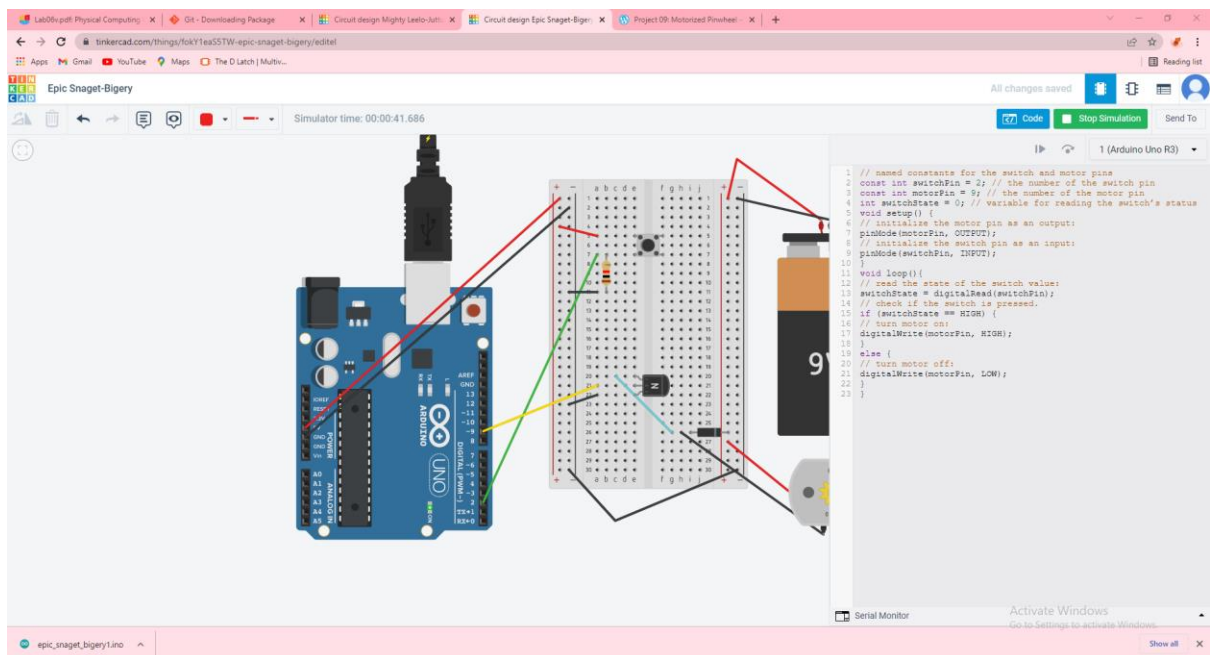
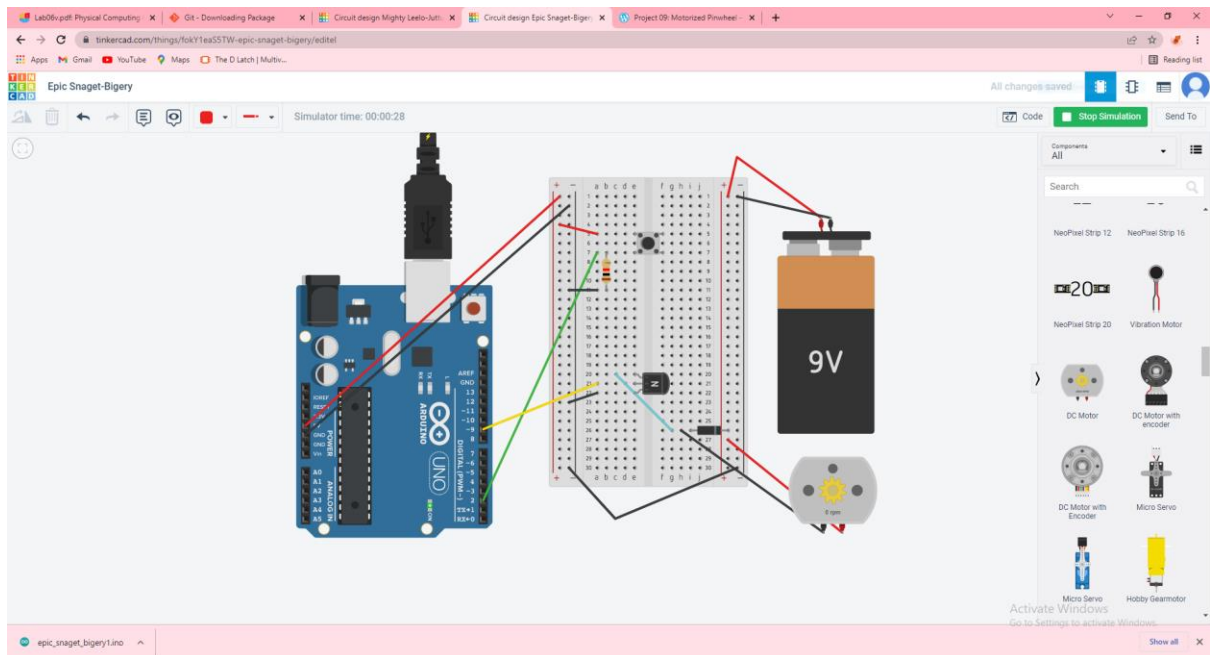
Date: 14.03.2022

Group: COMP1D-Y

Task1:



Task2:



## Task3:

Task 3:-

$$\beta_F = \frac{I_C}{I_B} \quad (1)$$

from (1)

$$\beta_F = \frac{I_C}{I_B} = 100$$

$$\Rightarrow I_C = \beta_F I_B$$

$$= 1.955 \text{ A}$$

220Ω

5V

$V_B = I_B R$

$\Rightarrow 5 - I_B = I_B (220)$  (2)

$I_B = 43 / (220) \text{ V}$

$= .020$

## Task4:

Arduino Uno R3

9V

Code

```

1 // named constants for the switch and motor pins
2 const int switchPin = 2; // the number of the switch pin
3 const int motorPin = 9; // the number of the motor pin
4 const int potentiometerPin = 3; // potentiometer pin
5 int potentiometerReading = 0; // int to take potentiometer reading
6 int motorSpeed = 0; // int to get motor speed
7 int switchState = 0; // variable for reading the switch's status
8 void setup() {
9   // initialize the motor pin as an output:
10  pinMode(motorPin, OUTPUT);
11  pinMode(potentiometerPin, OUTPUT);
12   // initialize the switch pin as an input:
13  pinMode(switchPin, INPUT);
14 }
15 void loop() {
16   // read the state of the switch value:
17   switchState = digitalRead(switchPin);
18   potentiometerReading = analogRead(potentiometerPin);
19   motorSpeed = potentiometerReading * 0.2492468127; //255/1023
20   // check if the switch is pressed.
21   if (switchState == HIGH) {
22     // turn motor on:
23     analogWrite(motorPin, motorSpeed);
24   }
25   else {
26     // turn motor off:
27     digitalWrite(motorPin, LOW);
28   }
29 }
30

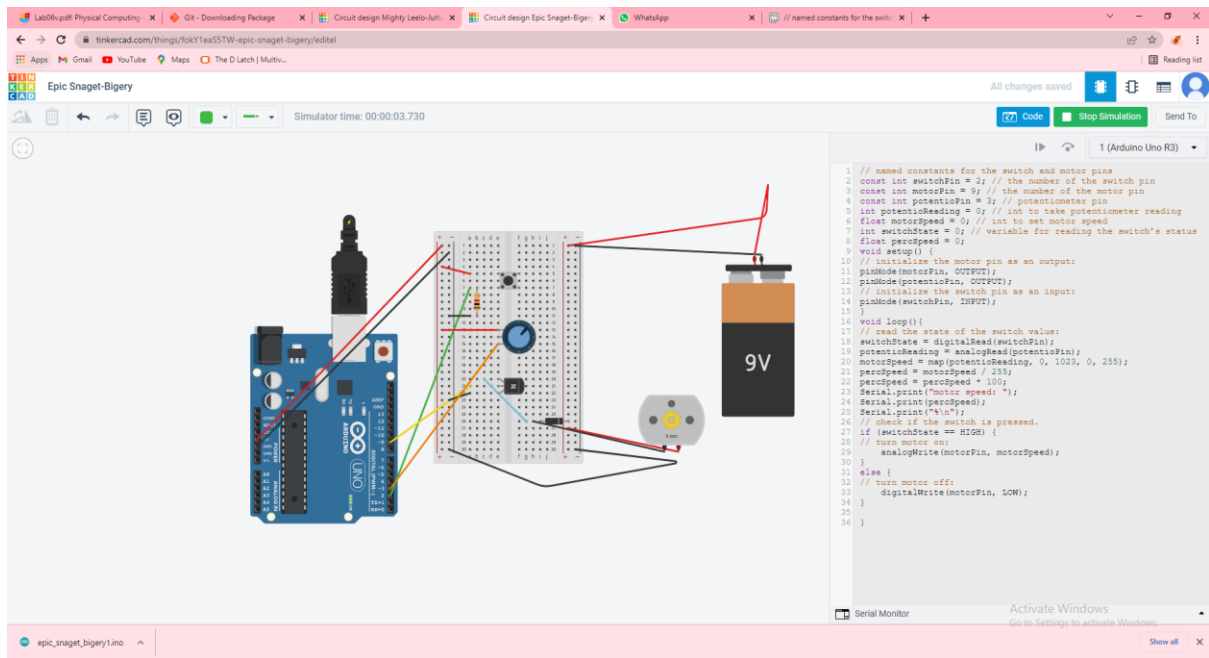
```

Serial Monitor

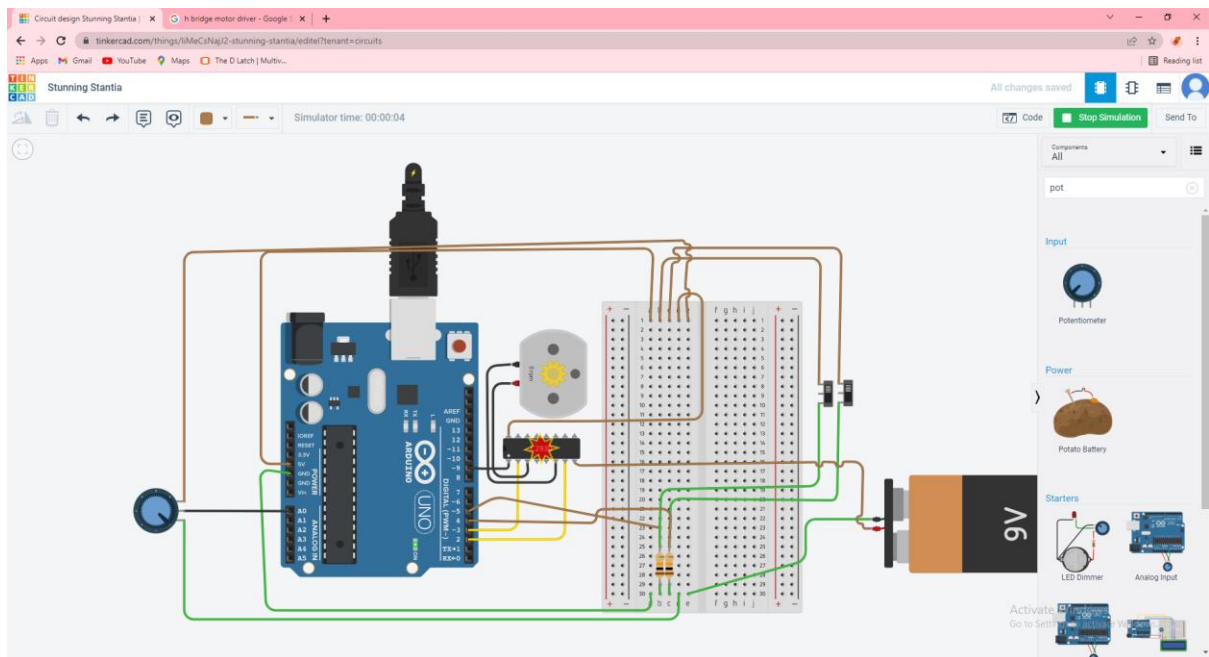
Activate Windows

Go to Settings to activate Windows.

## Task5:



## Task6:



Circuit design Stunning Stantia | X | h bridge motor driver - Google | X | +

linkencad.com/things/1M6CsNqJ2-stunning-stantia/edit?tenant=circuits

Apps | Gmail | YouTube | Maps | The D Latch | Multiv...

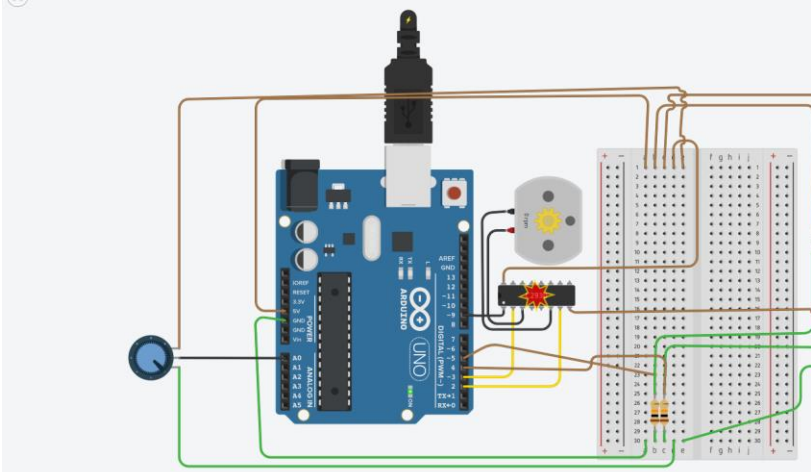
Stunning Stantia

Simulator time: 00:00:13

All changes saved

Code | Stop Simulation | Send To

1 (Arduino Uno R3)



```
1 const int controlPin1 = 3;
2 const int controlPin2 = 3;
3 const int enablePin = 0;
4 const int directionSwitchPin = 4;
5 const int onOffSwitchStateSwitchPin = 5;
6 const int potPin = A0;
7 // create some variables to hold values from your inputs
8 int onOffSwitchState = 0;
9 int previousOnOffSwitchState = 0;
10 int directionSwitchState = 0;
11 int previousDirectionSwitchState = 0;
12 int motorEnabled = 0; // Turns the motor on/off
13 int motorSpeed = 0; // speed of the motor
14 int motorDirection = 1; // current direction of the motor
15
16 void setup() {
17   // initialize the inputs and outputs
18   pinMode(directionSwitchPin, INPUT);
19   pinMode(onOffSwitchStateSwitchPin, INPUT);
20   pinMode(controlPin1, OUTPUT);
21   pinMode(controlPin2, OUTPUT);
22   pinMode(enablePin, OUTPUT);
23   // pull the enable pin LOW to start
24   digitalWrite(enablePin, LOW);
25 }
26
27 void loop() {
28   // read the value of the on/off switch
29   onOffSwitchState = digitalRead(onOffSwitchStateSwitchPin);
30   delay(1);
31   // read the value of the direction switch
32   directionSwitchState = digitalRead(directionSwitchPin);
33   // read the value of the pot and divide by 4 to get
34   // a value that can be used for PWM
35   motorSpeed = analogRead(potPin) / 4;
36   // if the on/off button changed state since the last loop()
37   if (onOffSwitchState != previousOnOffSwitchState) {
38     // change the value of motorEnabled if pressed
39     if (onOffSwitchState == HIGH) {
40       motorEnabled = !motorEnabled;
41     }
42   }
43   // if the direction button changed state since the last loop()
44   if (directionSwitchState != previousDirectionSwitchState) {
45     // change the value of motorDirection if pressed
46     if (directionSwitchState == HIGH) {
47       motorDirection = !motorDirection;
48     }
49   }
50   // activate the motor
51   digitalWrite(enablePin, HIGH);
52   analogWrite(controlPin1, motorSpeed);
53   analogWrite(controlPin2, motorSpeed);
54   digitalWrite(controlPin1, motorDirection);
55   digitalWrite(controlPin2, !motorDirection);
56 }
```

Serial Monitor