EE324 : Controls Lab
Lab 1 Report
DC Motor Position Control using Arduino

Mitul Wankhede , 20D070051
Mohit , 20D070052
Monu Kumar Yadav , 20D070053

September 5, 2022

Aim:

To design, implement and test a PID system to control the position of a DC Motor using Arduino Mega.

Objectives:

1. To rotate the motor by precisely 180 degrees with respect to its current position.
2. To model the task to satisfy specific design constraints of Rise time, Settling time and Percent overshoot.

Equipment used:

DC Motor, Arduino Mega board, L293D IC, Wires, Power Supply, Breadboard, Screwdriver, Wire stripper.

Methods:

Basic Block diagram for connections:

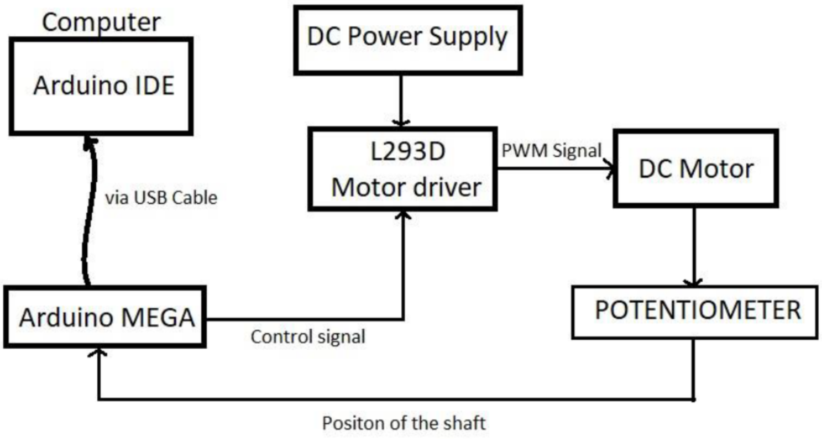


Figure 1: Block Diagram

Pin Diagram for L293D IC:

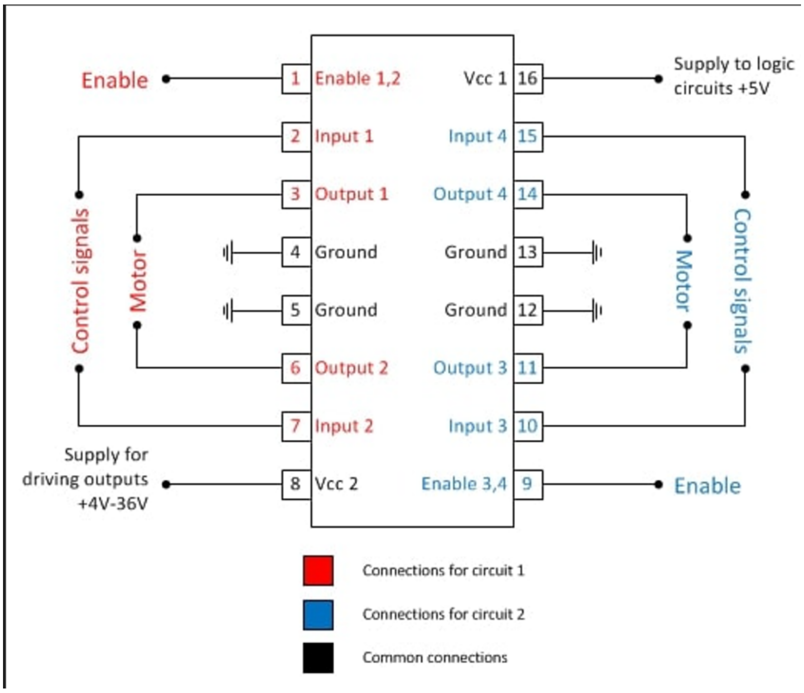
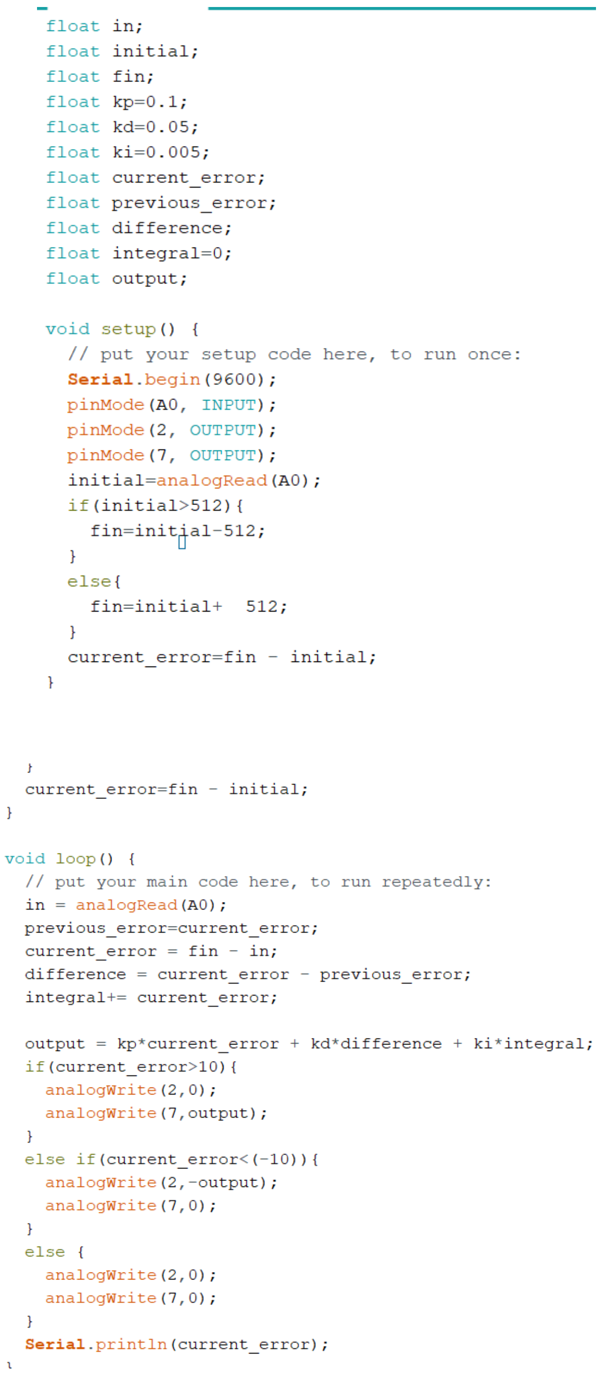


Figure 2: L293D pin diagram

Procedure:

- First the Potentiometer values were measured while rotating the motor by hand. This gave us a map between Angle, Pot values and Arduino variable value.
- The Arduino Board is connected to the PC which will upload the Arduino code.
- The board is in turn, connected to the L293D IC, which controls the Motor.
- Thus, The Control Signals (pin 2 & 7) of L293D are connected to the Arduino Board.
- The Motor pins (pin 3 & 7) are connected to the motor supply.
- The Motor Potentiometer is fed back to the Arduino board, and the voltage-to-angle calibrations are done in the Arduino code itself.
- The Enable and Supply pins of L293D are connected to each other.
- Pin 8 of L293D is given a 12V Supply to match the motor supply.

Arduino Code:

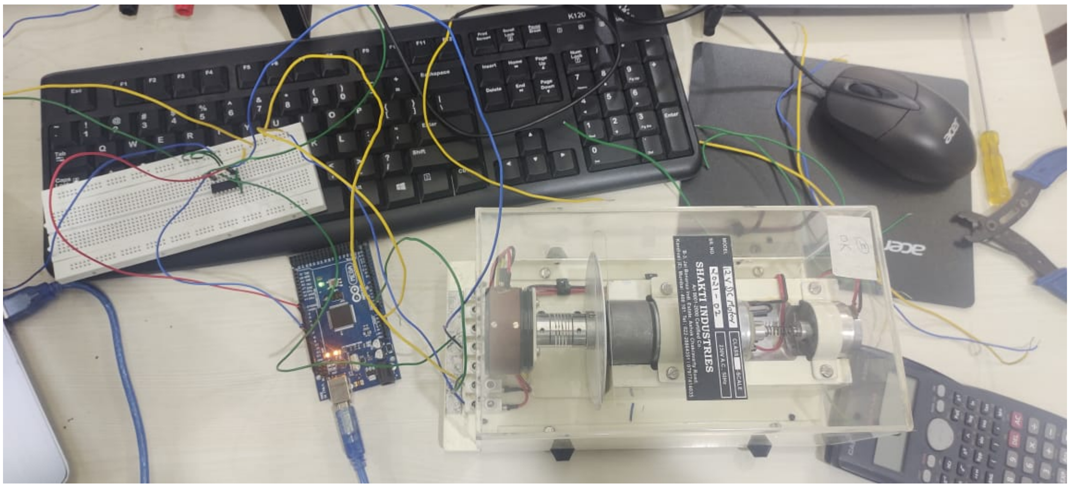


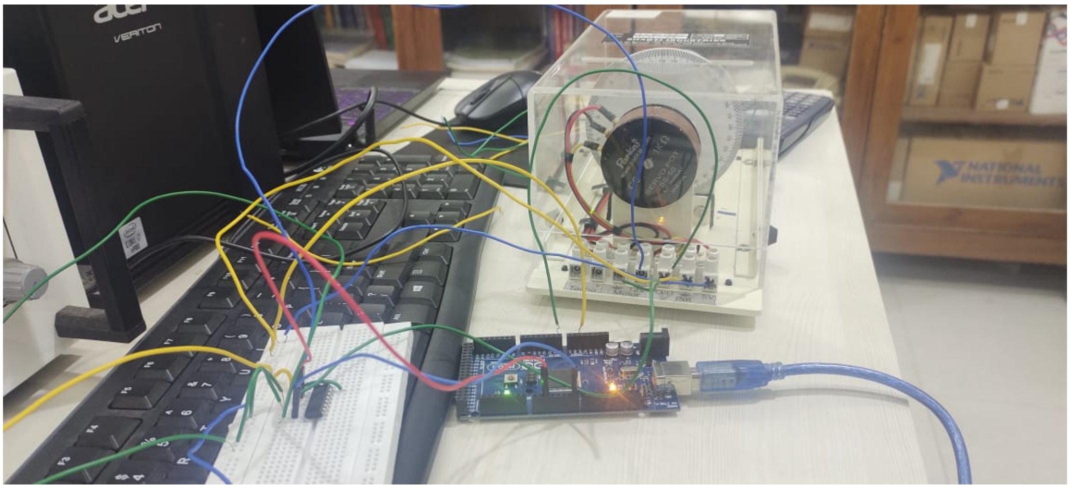
```
float in;
float initial;
float fin;
float kp=0.1;
float kd=0.05;
float ki=0.005;
float current_error;
float previous_error;
float difference;
float integral=0;
float output;

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  pinMode(A0, INPUT);
  pinMode(2, OUTPUT);
  pinMode(7, OUTPUT);
  initial=analogRead(A0);
  if(initial>512){
    fin=initial-512;
  }
  else{
    fin=initial+  512;
  }
  current_error=fin - initial;
}
```

```
  }
  current_error=fin - initial;
}

void loop() {
  // put your main code here, to run repeatedly:
  in = analogRead(A0);
  previous_error=current_error;
  current_error = fin - in;
  difference = current_error - previous_error;
  integral+= current_error;

  output = kp*current_error + kd*difference + ki*integral;
  if(current_error>10){
    analogWrite(2,0);
    analogWrite(7,output);
  }
  else if(current_error<(-10)){
    analogWrite(2,-output);
    analogWrite(7,0);
  }
  else {
    analogWrite(2,0);
    analogWrite(7,0);
  }
  Serial.println(current_error);
}
```

Figure 3: Code Snippet

Connections:





Observations:

We observe that our setup is successful in rotating the Motor by 180 Degrees to a reasonable margin of error, in both directions.

Now in order to meet the Control Constraints, variables K_p, K_d and K_i will be varied in the code and results will be noted.

The constraints to achieve are:

- Rise Time = 0.5 seconds
- Settling time = 1 seconds
- Percentage Overshoot = 10%

After varying and fine tuning the Control parameters, the following Control Statistics were achieved:

- Rise Time = 0.485 seconds
- Settling time = 0.982 seconds
- Percentage Overshoot = 2.56%

The values of the Control Parameters were:

$$K_p = 0.1,\ K_i = 0.005,\ K_d = 0.05$$

Thus, the required Control constraints were achieved.

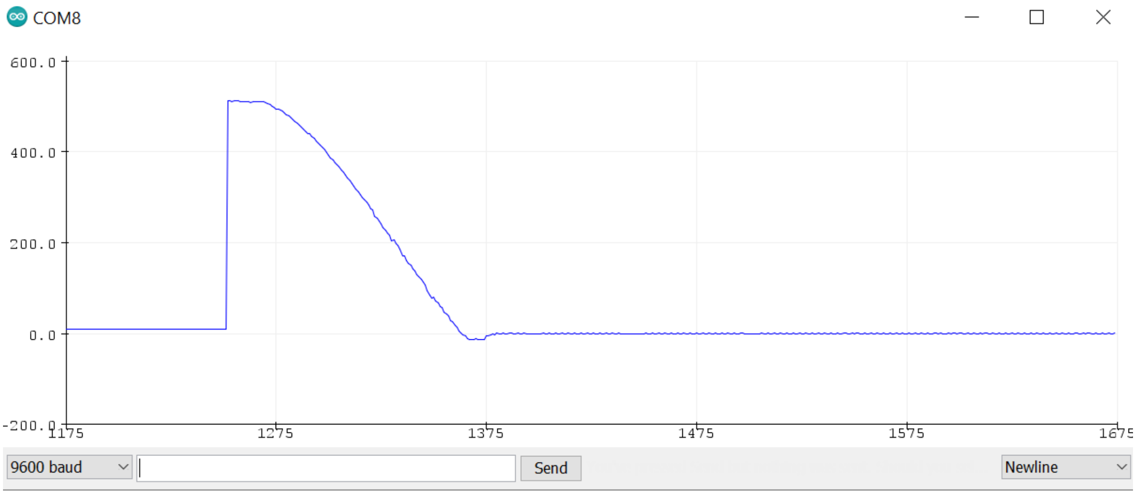


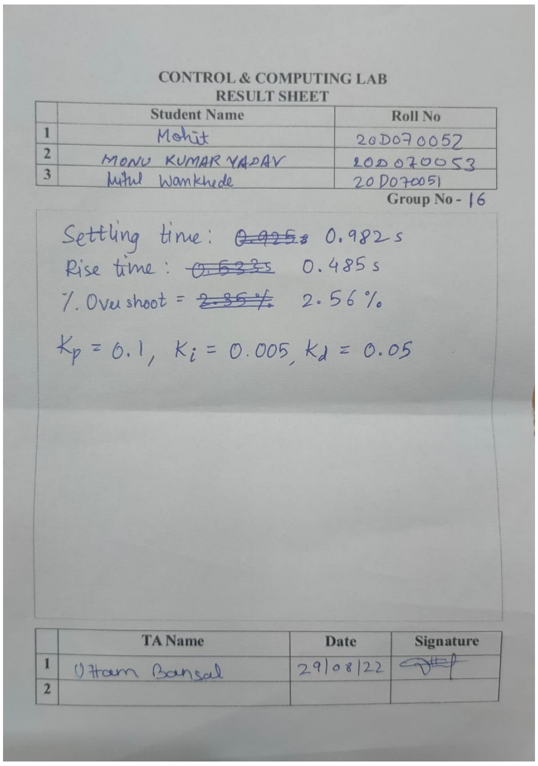
Figure 4: Position Graph

Problems Faced:

- Arduino Coding was to be learnt from scratch.
- Ground connections of various components were not connected which led to incorrect value.
- The Motor had a 'Nonlinear Region' : The Potentiometer values were random in a certain small region. This region was avoided during testing.
- The margin of error was large which led to the motor rotating more or less than 180 degrees so the margin was decreased.
- The values of K_p, K_i and K_d had to be randomly varied and observe there effects in order to meet the Design Constraints.
- While first measuring the times there were some offset in values but then we observed the supply voltage of 12V has dropped to 11V somehow so we took the measurements again after making it to 12V.

Experiment Completion Status:

The experiment has been completed and submitted in 3 Lab sessions.



CONTROL & COMPUTING LAB
RESULT SHEET

	Student Name	Roll No
1	Mohit	20D070052
2	MONU KUMAR YADAV	20D070053
3	Mitul Wankhede	20D070051

Group No - 16

Settling time: ~~0.925s~~ 0.982 s

Rise time: ~~0.533s~~ 0.485 s

% Overshoot = ~~2.35%~~ 2.56%

$K_p = 0.1$, $K_i = 0.005$, $K_d = 0.05$

	TA Name	Date	Signature
1	Uttam Bansal	29/08/22	
2			