PROJECT REPORT

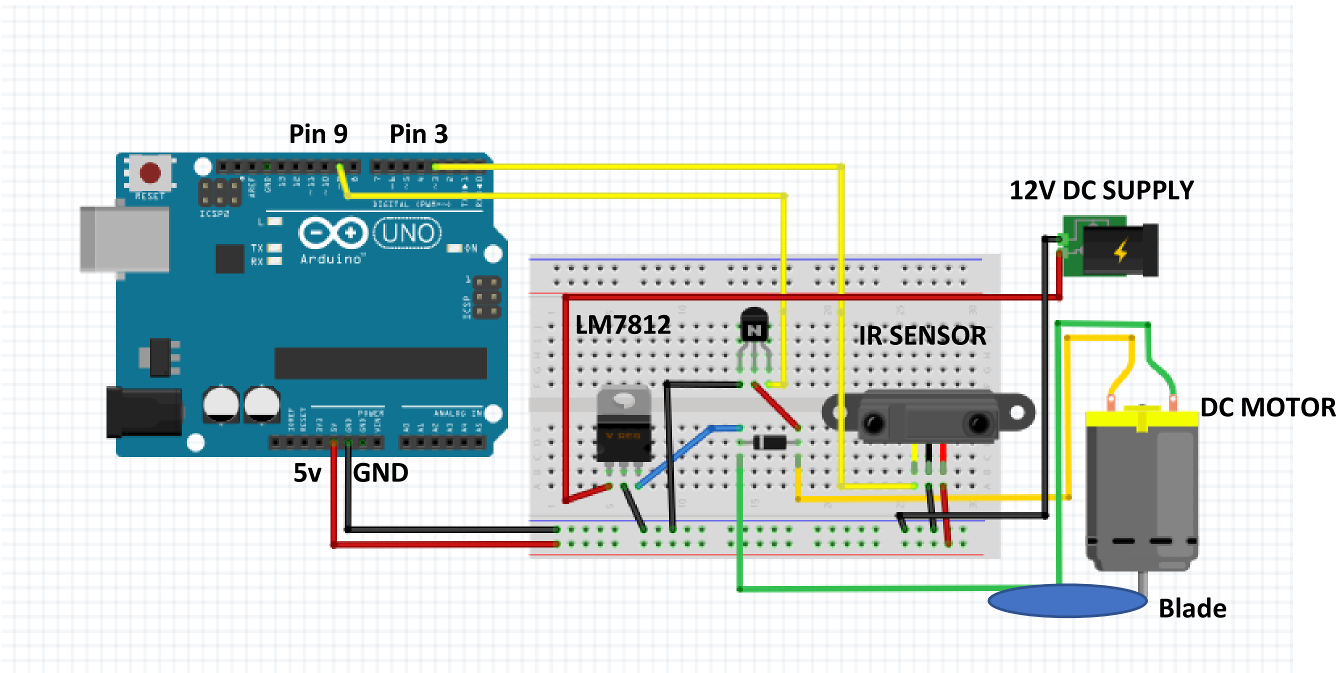
CLOSED LOOP SPEED CONTROL OF DC MOTOR

Problem Statement:

To design a closed loop speed control system for a 12V DC motor using Arduino.

This setup makes the motor spin at a desired rpm and does not allow any load to affect it.

Experimental Setup:



Methodology:

Module 1: To calculate the rpm of a dc motor.

Revolutions per minute (rpm):

Rpm is defined as the number of rotations in one minute. The rpm of a motor is calculated by building a tachometer.

Tachometer:

The tachometer is designed using IR sensor module. As IR sensor transmits IR rays which reflect back to IR receiver on detection of the object/shaft, the IR Module generates a pulse which is detected by the arduino controller.

$$duration = current\ time - previous\ time$$

$$t = 60 \times 1000ms = 1\ minute$$

$$current\ rpm = \frac{t}{duration}$$

where,

$current\ time - previous\ time$ is time between simultaneous detection of the shaft by the infrared sensor.

t is the time in seconds as Arduino assigns time in milliseconds.

Rpm is hence calculated by the above formula.

Module 2: To make the dc motor run at a desired speed.

Desired rpm:

The desired rpm is given by the user through the serial monitor.

Closed loop negative feedback control system:

A feedback system is one in which the output signal is sampled and then fed back to the input to form an error signal that drives the system.

A **set point** is the desired value for an essential variable or the process value of a system.

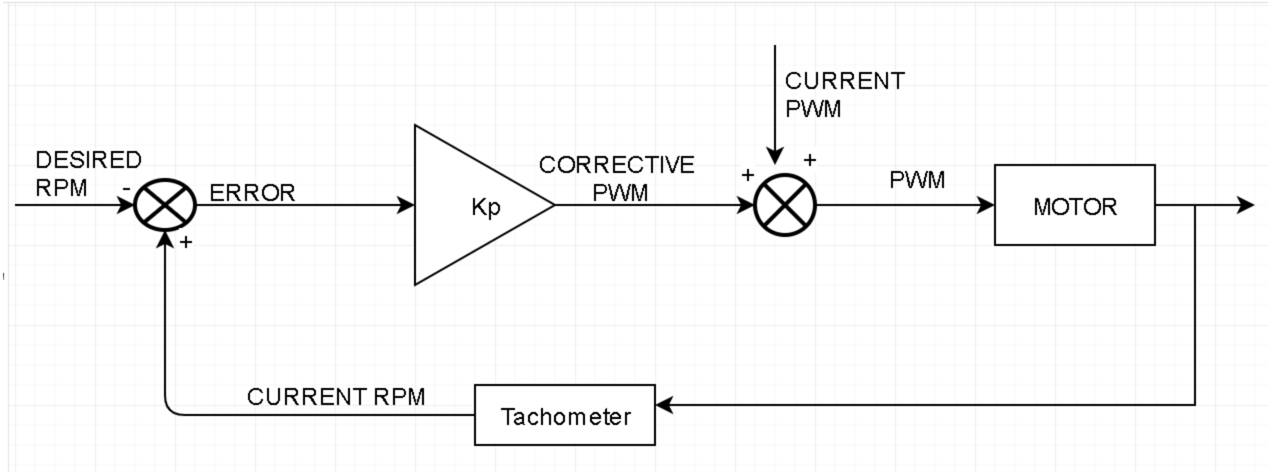
In a "negative feedback control system", the set point and current output values are subtracted from each other to obtain error.

Pulse width modulation (PWM):

The main purpose of Pulse-width modulation (PWM) is to control the voltage that is supplied to the DC motor.

Proportional Controller:

It is a negative feedback control system in which a correction term, which is proportional to the error, is applied to the control variable.



$$error = current\ rpm - desired\ \ rpm$$

$$corrective\ pwm = error\ \times Kp$$

$$pwm = corrective\ pwm + pwm$$

where Kp is the proportionality gain.

Permissible error:

The permissible error of a system is the maximum tolerance of setpoint error.

It can be defined by taking the least count of the system.

$$\frac{rpm(max)}{pwm(max)}$$

Module 3:

Load stabilization:

Load is the external force exerted on the shaft of the motor.

Load stabilization refers to the ability of the motor to regain its speed after an unexpected load is applied to the motor.

Algorithm:

1. Initially the user enters the desired speed in rpm and the current rpm of the motor is calculated using the tachometer.
2. The difference between current rpm and desired rpm is calculated and stored as **"Error".**

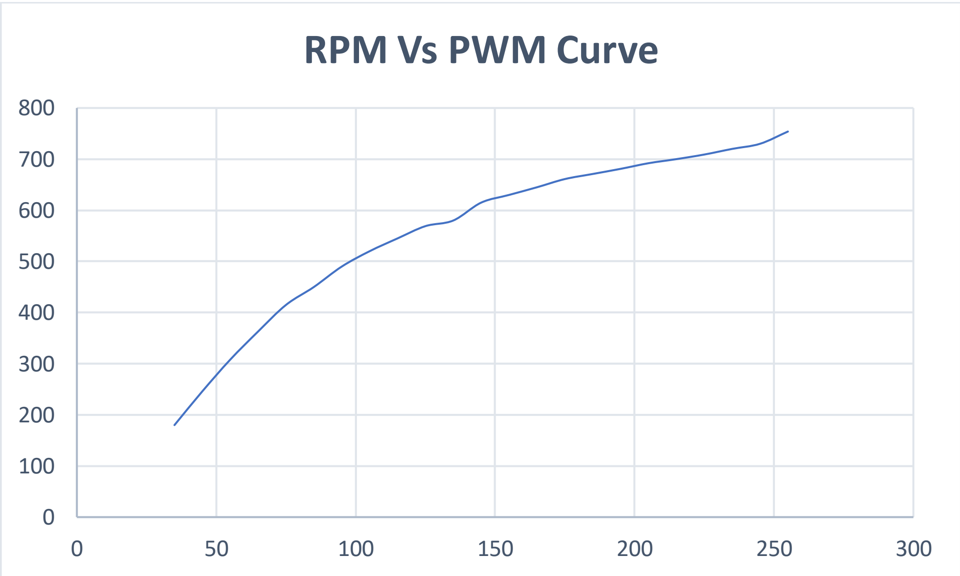
3. This error is further passed on to the proportional controller to obtain the corrective pwm which is added to current pwm to give resultant pwm.
4. This resultant pwm is given as input to analogwrite() to run the motor .
5. Steps 2, 3 and 4 are repeated until the error is within the permissible range.(refer module 2)

Practical Results:

Rpm Vs Pwm Curve:

On taking the various values of rpm to its equivalent pwm value, an rpm Vs pwm curve is plotted.

Graphical Representation:



Value of proportional gain constant Kp:

- ***Kp=0.33***

The rpm Vs pwm curve is approximated to be a linear graph and its slope is calculated.

$$Kp = \frac{pwm(max)}{rpm(max)}$$

Although the total average time period = 2801.88ms, the settling time of motor for error correction above 500 is witnessed to take longer duration.

Due to the above drawbacks, the value of Kp was found experimentally by varying the constants to arbitrary values below 1.

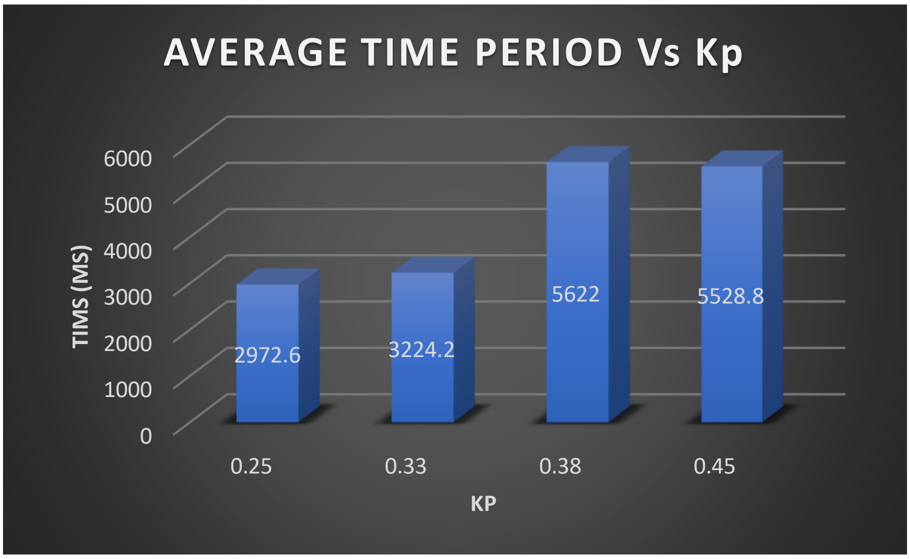
Comparison Table:

A comparison was made with the different Kp values that were taken.

rpm	**200**	**200 to 300**	**300 to 400**	**400 to 500**	**500 to 600**	**600 to 700**
Kp						
0.25	0	4821	4366	2298	1813	1565
0.33	0	6120	3984	3001	1786	1230
0.38	0	10736	8648	5150	2495	1081
0.45	0	13622	6658	3926	1855	1583

A error difference of 100 was taken in order to observe broader variations in rpm.

Average time period for an error of 100 Vs Kp:



- ***Kp=0.25***

  When Kp was taken as 0.25, the results obtained were improved with respect to settling time and stabilisation.

  Hence the practical value of proportional gain constant Kp = 0.25

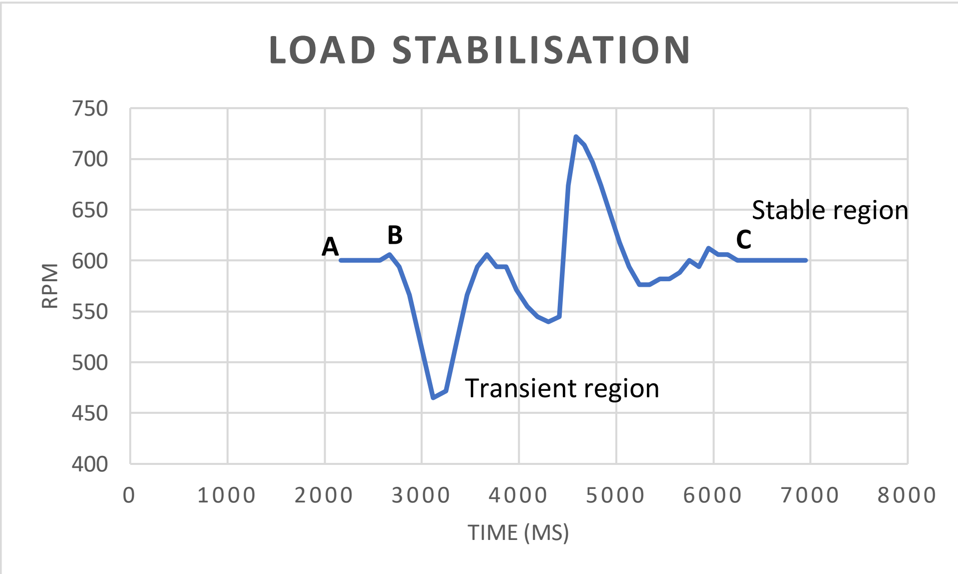
RPM Vs Time(ms) Matrix for Kp=0.25 :

FROM (rpm)	200	300	400	500	600	700
TO (rpm)						
200	0	5052	4566	2974	4582	4696
300	4526	0	4002	4296	4110	4081
400	3998	3390	0	3853	3989	4525
500	2889	3145	2962	0	1868	2028
600	2254	2185	2162	1666	0	1104
700	1349	1319	1584	1344	1629	0

Average Time Period: 2559.11 ms

Load Stabilization:

Graphical Representation:



Inference:

Region A to B: This region refers to the stable state of the dc motor running at a speed of 600 rpm.

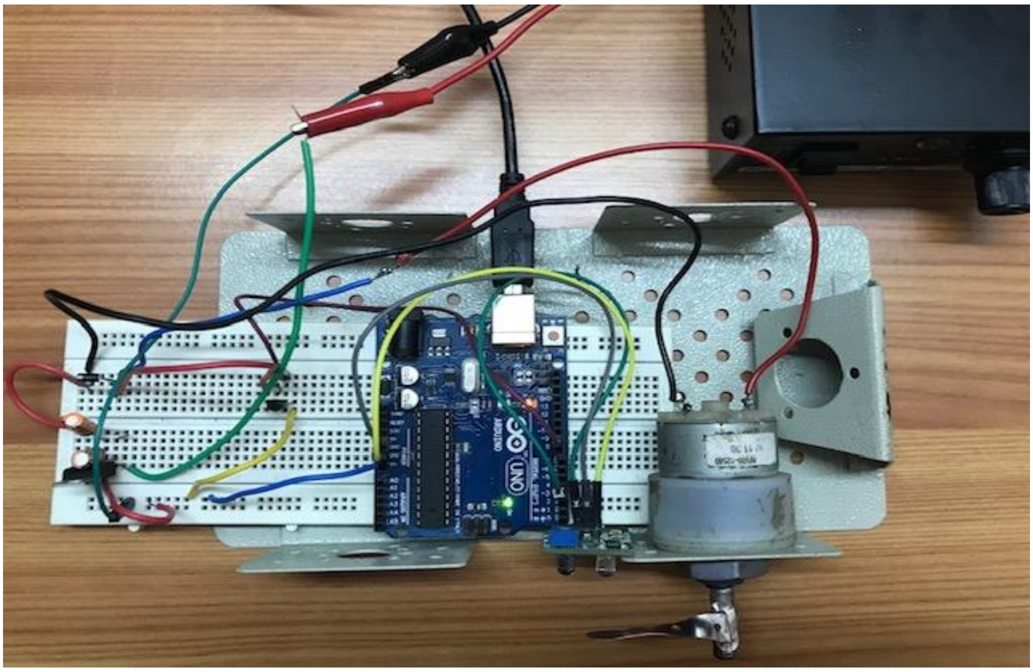
Region B to C: When a load is applied across a motor for **2 seconds**, the speed of the motor is found to fluctuate from its actual value.

Eventually the motor is seen to regain its original speed. This process is known as load stabilization.

The time taken for the dc motor to regain its speed is known as the **transient time** . The transient time is **3481ms.**

Appendix:

Working Model:



Learnings and outcomes:

This project has made me understand the practical usage of transistors and voltage regulators.

Gained the capacity to understand and apply theoretical topics to real life problems.

The major problem faced was to obtain an optimum settling time. This problem was solved by fine tuning of the proportional constant.