A

PROJECT REPORT ON

"Low Cost Customizable Motor Driver"

Submitted by,

Durgesh Kolte	**T196003**
Yash Pamnani	**T196011**

Guided by,

Shridhar Khandekar

A Report submitted to MIT Academy of Engineering, Alandi (D.) submitted in partial fulfilment of the requirement for Fifth Semester of BACHELOR OF TECHNOLOGY in School of Electrical Engineering.

MIT | Academy of Engineering

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

SCHOOL OF ELECTRICAL ENGINEERING

MIT Academy of Engineering

Dehu Phata, Alandi (D.)

Pune - 412105, Maharashtra (India)

2019-20

MIT | Academy of Engineering	**School of Electrical Engineering** **(Accredited by NBA, ISO 9001:2008 Certified)**

CERTIFICATE

This is certify that,

Durgesh Kolte	**T196003**
Yash Pamnani	**T196011**

of T. Y. B. Tech. have submitted a Report on,

Low Cost Customizable Motor Driver

The said work is completed by putting the requirement of hours as per prescribed curriculum during the academic year 2019–20. The report is submitted in the partial fulfilment of the requirements for the course **Fundamentals of Robotics** in the Fifth Semester of Degree of Engineering in –School of Electrical Engineering, MIT Academy of Engineering.

Date : 29 / 11 / 2019

Place : MIT Academy of Engineering, Alandi (D.)

Signature:	Signature:
	Name :
Project Advisor	**Dean SEE**

CONTENTS

ACKNOWLEDGEMENT

We take this opportunity to record our profound gratitude and indebtedness to Mr. Shridhar Khandekar, Assistant Professor, School of Electrical Engineering for his inspiring guidance, valuable advises, constant encouragement and untiring supervision throughout our project work. He has been supportive in the whole designing and implementing phase of this research work and we are grateful for his help in whatsoever manner provided.

Finally, we would like to acknowledge and express our special thanks to our family, friends and classmates for their patience, encouragement, support they have made during the period of this work.

Durgesh Kolte	**T196003**
Yash Pamnani	**T196011**

ABSTRACT

Low Cost Customizable Motor Driver is a project based on Embedded Electronics being used in a variety of applications in various places which includes uses in places majorly of Robotics. A **Motor Driver** is capable of driving motors and by driving it also means controlling it according to our need or requirement. There are multiple manufacturers which supply motor drivers, but in general, Cytron and SmartElex Drivers are mostly used for high amperage applications as they are more accurate as compared to other currently available in market and available as well as require less complexity. In this project we have modelled our Motor Driver Design to work as in it can be available easily and be manufactured and customized in lowest cost possible.

There are various tools that help in performing this specific task of Simulating Motor Driver Circuits like Proteus, Multisim, etc. We have used Proteus, Arduino IDE, Eagle PCB Design as our helping Simulation software in our journey.

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION :

1.1 Motivation :

There are three major key aspects that are looked into by industries nowadays which are mainly cost, time and accuracy.

So whenever we talk about automation and robotics the main purpose of the company is to make maximum profit in less amount of time with minimum workforce and errors.

So this Motor Driver is designed such that it can be manufactured and customized at any simple industry or workspaces like Colleges, Schools, Workshop ventures, etc. with minimum cost and with the help of easily available components which makes them cheap for engineering students so that they can implement needed Drivers in their circuitry.

1.2 Objective :

To device a Motor Driver Circuit and implement it so that it could be easily manufactured and embedded over any custom local PCB for Robotics operation. The main objective of the Motor Driver is to be of :

1. Low Cost
2. Less Complexity
3. Easily available components

1.3 Problem Statement :

To Design and Implement a Motor Driver Circuitry which is capable of controlling motor at low cost and be less complex for customization.

2. MARKET SURVEY :

While crafting of this Motor Driver, firstly a thorough Research was made in the respective field about all the working and methodologies involved. After the Research work was over then finally the stage of Component Selection was carried out and each and every component which can fulfil our requirement was listed and components similar to our configuration were shorted out. MOSFETs in Market :

Fig No. 2.1 MOSFETs available in Market

3. TERMINOLOGY :

- **MOSFETs :**

A metal–oxide–semiconductor field-effect transistor (MOSFET, MOS-FET, or MOS FET) is a field-effect transistor (FET with an insulated gate) where the voltage determines the conductivity of the device. It is used for switching or amplifying signals. The ability to change conductivity with the amount of applied voltage can be used for amplifying or switching electronic signals. MOSFETs are now even more common than BJTs (bipolar junction transistors) in digital and analog circuits.

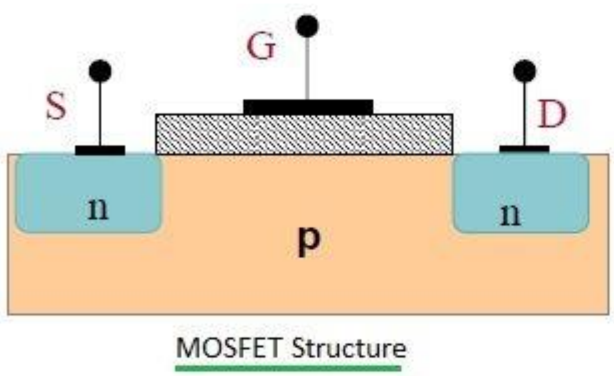

Fig No. 3.1 MOSFET structure

A MOSFET is by far the most common transistor in digital circuits, as hundreds of thousands or millions of them may be included in a memory chip or microprocessor. Since they can be made with either p-type or n-type semiconductors, complementary pairs of MOS transistors can be used to make switching circuits with very low power consumption, in the form of CMOS logic.

Why MOSFET?

MOSFETs are particularly useful in amplifiers due to their input impedance being nearly infinite which allows the amplifier to capture almost all the incoming signal. The main advantage is that it requires almost no input current to control the load current, when compared with bipolar transistors. MOSFETs are available in two basic forms:

- **Depletion Type:** The transistor requires the Gate-Source voltage (V_{GS}) to switch the device "OFF". The depletion mode MOSFET is equivalent to a "Normally Closed" switch.
- **Enhancement Type:** The transistor requires a Gate-Source voltage(V_{GS}) to switch the device "ON". The enhancement mode MOSFET is equivalent to a "Normally Open" switch.

MOSFET structure

It is a four-terminal device with source(S), gate (G), drain (D) and body (B) terminals. The body is frequently connected to the source terminal, reducing the terminals to three. It works by varying the width of a channel along which charge carriers flow (electrons or holes).

The charge carriers enter the channel at source and exit via the drain. The width of the channel is controlled by the voltage on an electrode is called gate which is located between source and drain. It is insulated from the channel near an extremely thin layer of metal oxide. A metal-insulator-semiconductor field-effect transistor or MISFET is a term almost synonymous with MOSFET. Another synonym is IGFET for insulated-gate field-effect-transistor.

MOSFET Operation

The working of a MOSFET depends upon the MOS capacitor. The MOS capacitor is the main part of MOSFET. The semiconductor surface at the below oxide layer which is located between source and drain terminals. It can be inverted from p-type to n-type by applying positive or negative gate voltages.

When we apply positive gate voltage the holes present under the oxide layer with a repulsive force and holes are pushed downward with the substrate. The depletion region populated by the bound negative charges which are associated with the acceptor atoms. The electrons reach channel is formed. The positive voltage also attracts electrons from the n+ source and drain regions into the channel. Now, if a voltage is applied between the drain and source, the current flows freely between the source and drain and the gate voltage controls the electrons in the channel. If we apply negative voltage, a hole channel will be formed under the oxide layer.

P-Channel MOSFET

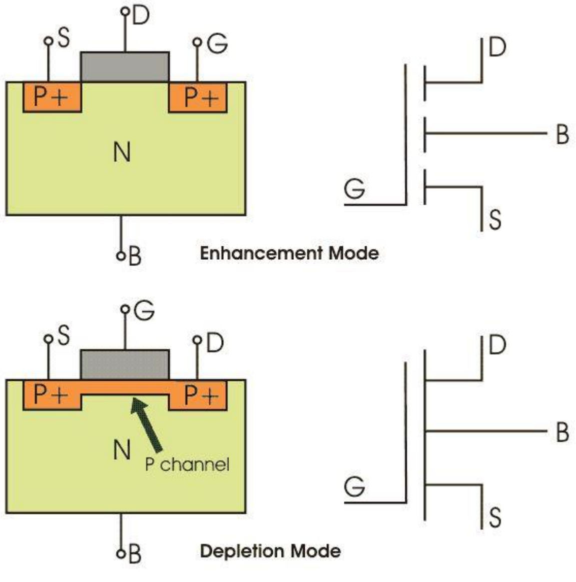

Fig No. 3.2 P-channel MOSFET

The drain and source are heavily doped p+ region and the substrate is in n-type. The current flows due to the flow of positively charged holes also known as p-channel MOSFET. When we apply negative gate voltage, the electrons present beneath the oxide layer experience repulsive force and they are pushed downward in to the substrate, the depletion region is populated by the bound positive charges which are associated with the donor atoms. The negative gate voltage also attracts holes from p+ source and drain region into the channel region.

N-Channel MOSFET

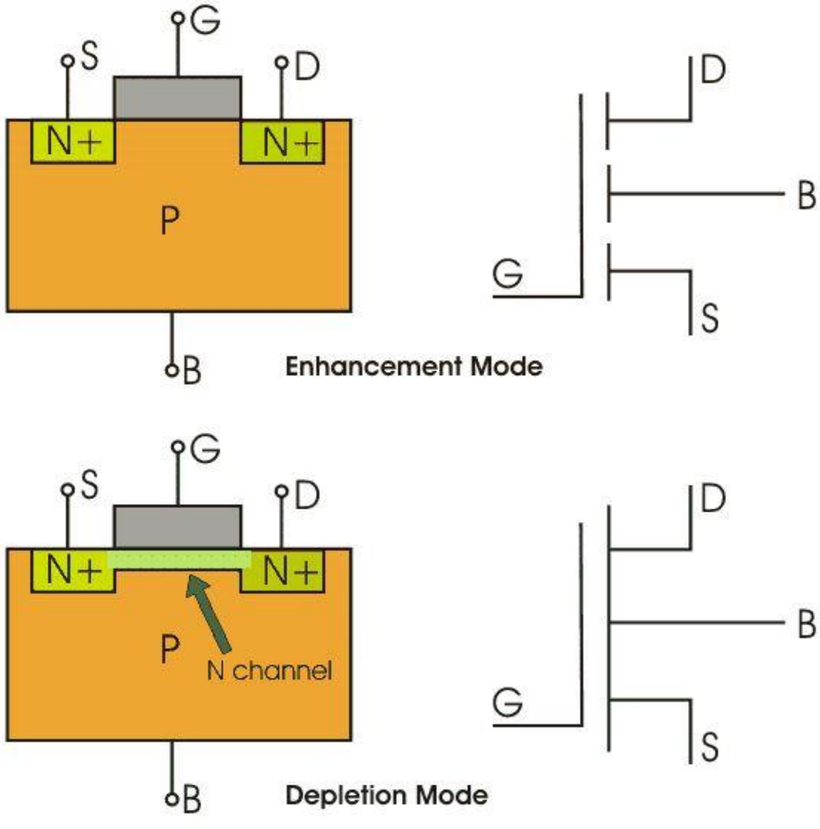

Fig No. 3.3 N-channel MOSFET

The drain and source are heavily doped n+ region and the substrate is p-type. The current flows due to the flow of negatively charged electrons, also known as n-channel MOSFET. When we apply the positive gate voltage the holes present beneath the oxide layer experience repulsive force and the holes are pushed downwards in to the bound negative charges which are associated with the acceptor atoms. The positive gate voltage also attracts electrons from n+ source and drain region in to the channel thus an electron reach channel is formed. [1]

- **Frequency :**

Frequency is the number of occurrences of a repeating event per unit of time. It is also referred to as temporal frequency, which emphasizes the contrast to spatial frequency and angular frequency. The period is the duration of time of one cycle in a repeating event, so the period is the reciprocal of the frequency. [2]

- **PWM :**

What is Pulse-width Modulation?

Pulse Width Modulation (PWM) is a fancy term for describing a type of digital signal. Pulse width modulation is used in a variety of applications including sophisticated control circuitry. A common way we use them here at SparkFun is to control dimming of RGB LEDs or to control the direction of a servo motor. We can accomplish a range of results in both applications because pulse width modulation allows us to vary how much time the signal is high in an analog fashion. While the signal can only be high (usually 5V) or low (ground) at any time, we can change the proportion of time the signal is high compared to when it is low over a consistent time interval.

Suggested Reading

Some background tutorials you might consider first:

- Voltage, Current, Resistance, and Ohm's Law
- Analog vs Digital
- Voltage Dividers
- Digital Logic

Duty Cycle

When the signal is high, we call this "on time". To describe the amount of "on time" , we use the concept of duty cycle. Duty cycle is measured in percentage. The percentage duty cycle specifically describes the percentage of time a digital signal is on over an interval or period of time. This period is the inverse of the frequency of the waveform.

If a digital signal spends half of the time on and the other half off, we would say the digital signal has a duty cycle of 50% and resembles an ideal square wave. If the percentage is higher than 50%, the digital signal spends more time in the high state than the low state and vice versa if the duty cycle is less than 50%. 100% duty cycle would be the same as setting the voltage to 5 Volts (high). 0% duty cycle would be the same as grounding the signal.Here is a graph that illustrates these three scenarios: [3]

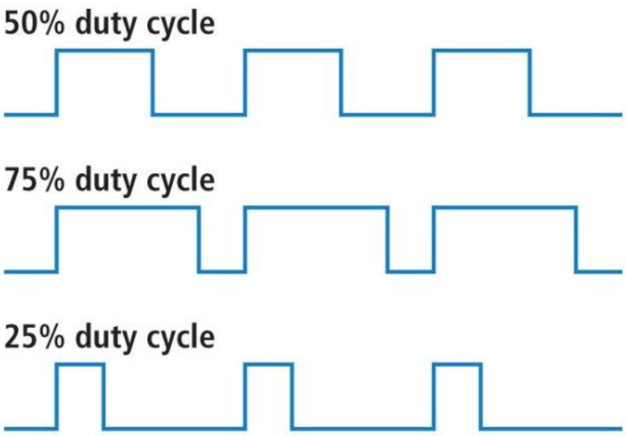

Fig No. 3.4 50%, 75%, and 25% Duty Cycle Examples

- **Amplitude :**

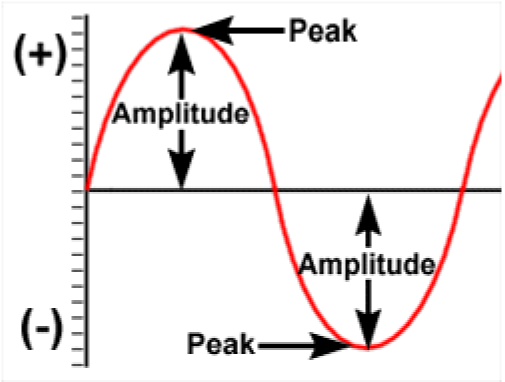

Fig No. 3.5 Amplitude Graph

The first characteristic of AC power is its "amplitude". Amplitude is the maximum value of current or voltage. It is represented by either of the two peaks of the since wave. This voltage level is also referred to as the peak voltage, and can be either positive or negative. Positive and negative refer only to the direction of current flow. A negative number does not mean that the voltage or current flow are less than zero, only that the current flows in the opposite direction. [4]

4. METHODOLOGY :

- **H-Bridge :**

An H bridge is an electronic circuit that switches the polarity of a voltage applied to a load. These circuits are often used in robotics and other applications to allow DC motors to run forwards or backwards.

Most DC-to-AC converters (power inverters), most AC/AC converters, the DC-to-DC push–pull converter, most motor controllers, and many other kinds of power electronics use H bridges. In particular, a bipolar stepper motor is almost invariably driven by a motor controller containing two H bridges.

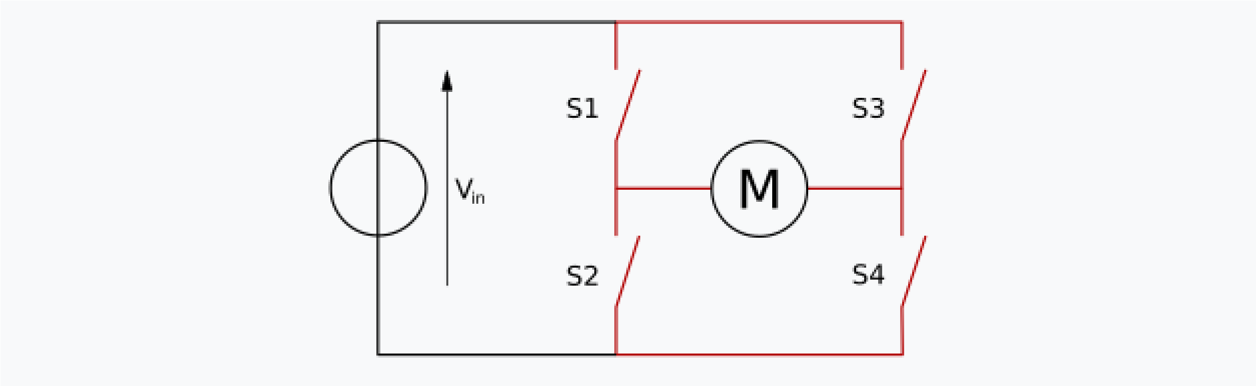

Fig No. 4.1 Structure of an H bridge (highlighted in red)

H bridges are available as integrated circuits, or can be built from discrete components. [6]

The term *H bridge* is derived from the typical graphical representation of such a circuit. An H bridge is built with four switches (solid-state or mechanical). When the switches S1 and S4 (according to the first figure) are closed (and S2 and S3 are open) a positive voltage will be applied across the motor. By opening S1 and S4 switches and closing S2 and S3 switches, this voltage is reversed, allowing reverse operation of the motor.

Using the nomenclature above, the switches S1 and S2 should never be closed at the same time, as this would cause a short circuit on the input voltage source. The same applies to the switches S3 and S4. This condition is known as shoot-through.

Operation

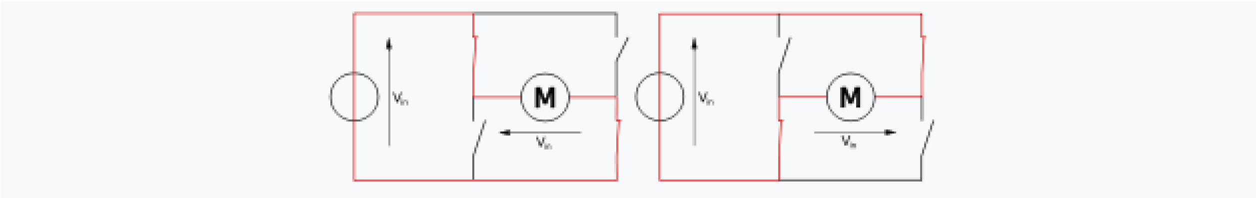

Fig No. 4.2 Basic states of an H bridge

The H-bridge arrangement is generally used to reverse the polarity/direction of the motor, but can also be used to 'brake' the motor, where the motor comes to a sudden stop, as the motor's terminals are shorted, or to let the motor 'free run' to a stop, as the motor is effectively disconnected from the circuit. The following table summarises operation, with S1-S4 corresponding to the diagram above. [5]

S1	S2	S3	S4	Result
1	0	0	**1**	Motor moves right
0	**1**	**1**	0	Motor moves left
0	0	0	0	
1	0	0	0	
0	**1**	0	0	Motor coasts
0	0	**1**	0	
0	0	0	**1**	
0	**1**	0	**1**	Motor brakes
1	0	**1**	0	
x	x	**1**	**1**	Short circuit
1	**1**	x	x	

Table No. 4.1 Switch Operations

5. PLANNING :

5.1 Component Selection :

Important parameters for selecting a **MOSFET** :

It is important to consider following parameters included in datasheet before choosing a MOSFET

- **Id** : This is the max total current that can pass through the load
- **Vdss** : Max voltage that can be connected to load
- **Vgs** : Max gate to source voltage without getting burned
- **Vgs (threshold)** : min voltage to switch the MOSFET
- **rDS (on)** : resistance offered when current passes through drain and source

- **Applying minimum Vgs to switch the MOSFETs :**

Applying the threshold voltage does not mean you are ready to use the MOSFET if you look at the datasheet you’ll see that at its threshold voltage only a certain minimum current can flow through it.

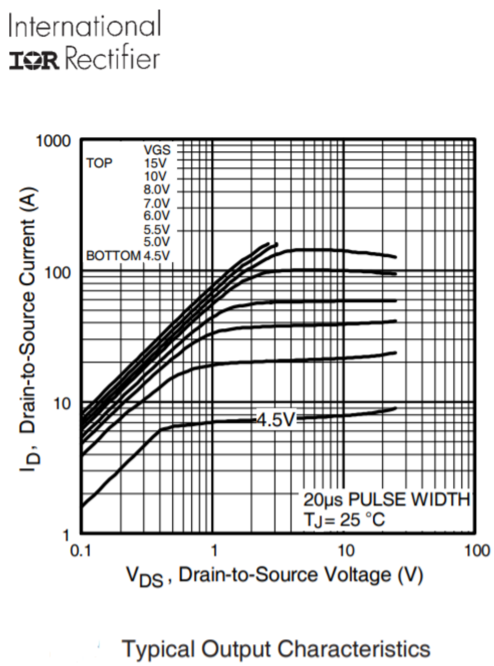

Fig No. 5.1 Output Characteristics

For bigger loads that draws more current, you'll have to apply more voltage you'll need to apply more Vgs to enhance the drain to source channel.

- **Selection of MOSFETs :**

N-channel power MOSFET

IRFZ44	**FQP55nf06**
Vdss = 55V	Vdss = 60V
rDS = 0.0175 ohm	rDS = 0.015 ohm
Id = 49A	Id= 55A

Table No. 5.1 N-channel power MOSFET

P-channel power MOSFET

IRF4905	**FQP47P06**
Vdss = - 55V	Vdss= - 60V
rDS = 0.02 ohm	rDS = 0.026 ohm
Id = -74A	Id= -47A

Table No. 5.2 P-channel power MOSFET

N-channel signal MOSFET 2n7000	**Gate Driver TC4427**
Vdss = 60V	Vo = 4.5V – 18V
rDS = 5 ohm	
Id = 0.2A	Id= 1.5A

Table No. 5.3 MOSFET Driver

- **The Motor-Controller :**

The motor-controller is important because without one your robot won't move. You can use any type of motor-controller that can support the motors on your robot. The simplest of solid-state H-bridges can contain as few as four transistors, but limitations in PWM frequency and heat generated from cross-conduction quickly make this type of bridge unappealing. Adding a few extra parts to the simple four-transistor H-bridge can provide a versatile and inexpensive motor-controller.

We decided to build the H-bridge to meet the following minimum criteria as follows:

Continuous amperage = 12A

Voltage rating = 12V

Speed control capable (PWM)

PWM switching speed (32kHz)

- **High-Side Switches :**

The H-bridge uses P-channel MOSFETs for the high-side switches and N-channel MOSFETs for the low-side switches. Remember that a P-channel MOSFET (like a PNP transistor) is turned on by supplying a voltage to the Gate pin that is around 5–10v below the positive supply voltage, usually Ground.

To invert the signal required to turn the P-channel MOSFETs on, we used small N-channel signal MOSFETs to provide the GND supply to turn them on, whereas a 10k-ohm pull-up resistor (to V+) keeps them turned off when not used. The MOSFETs have tiny capacitors inside of them that must be drained each time it is switched in order to fully turn off. This is easily done with a pull-up or pull-down resistor. These MOSFETs are intended to be controlled using a digital output pin and are not set up to be driven using a PWM signal.

The P-channel MOSFETs are turned off by setting the Gate pin voltage equal to the Source pin, which is connected to the positive voltage supply. So, we place a 10k pull-up resistor from the Gate pin to the Source pin of each P-channel power MOSFET. This ensures that unless turned on, these MOSFETs will stay off.

- **Low-Side Switches :**

The N-channel MOSFETs on the low-side of the bridge are the logic-level type and can be driven directly by the Arduino, but the 40mA supplied by the Arduino PWM pins is hardly enough to drive the MOSFETs at higher PWM frequencies. To remedy this, we used a MOSFET driver chip that amplifies the Arduino PWM signal to supply around 2 amps to the MOSFET Gate pins—that's about 100 times more current than the Arduino can supply. Because of this, the N-channel MOSFETs can be driven at much higher frequencies without causing any problems.

The motor-controller schematic shown in the figure illustrates the operation of H-bridge in the circuit

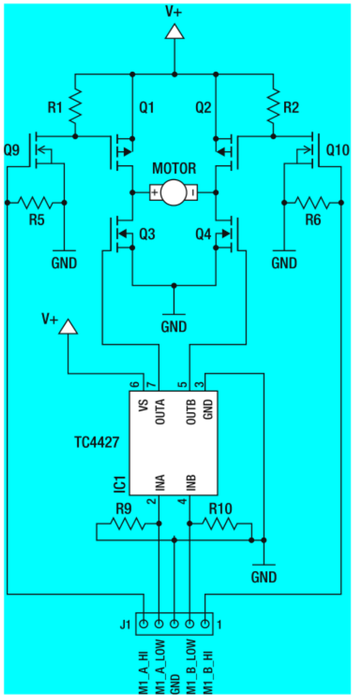

Fig No. 5.2 MOSFET Driver Circuit

The MOSFET driver uses the positive voltage supply from the batteries to power the N-channel MOSFET Gate pins, and the voltage limit of the TC4427 MOSFET driver IC is only 18v, so the maximum voltage of this circuit is 18vdc.

- **Use of MOSFET Driver TC4427** :

When the signal from the Arduino is given to TC4427, the frequency of the output (31KHz) remains same. It just amplifies the voltage, from 5V to 12V that we had connected to the MOSFET driver.

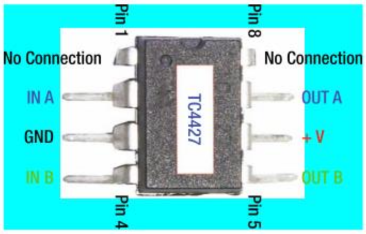

Fig No. 5.3 MOSFET Driver Pin-outs (TC4427)

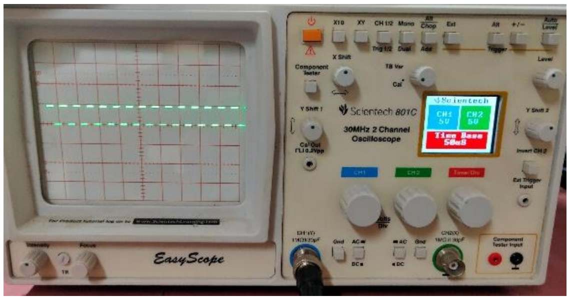

Fig No. 5.4 Arduino 5V, 31KHz without TC4427

Fig No. 5.5 Arduino 5V, 31KHz with TC4427

5.2 Simulations :

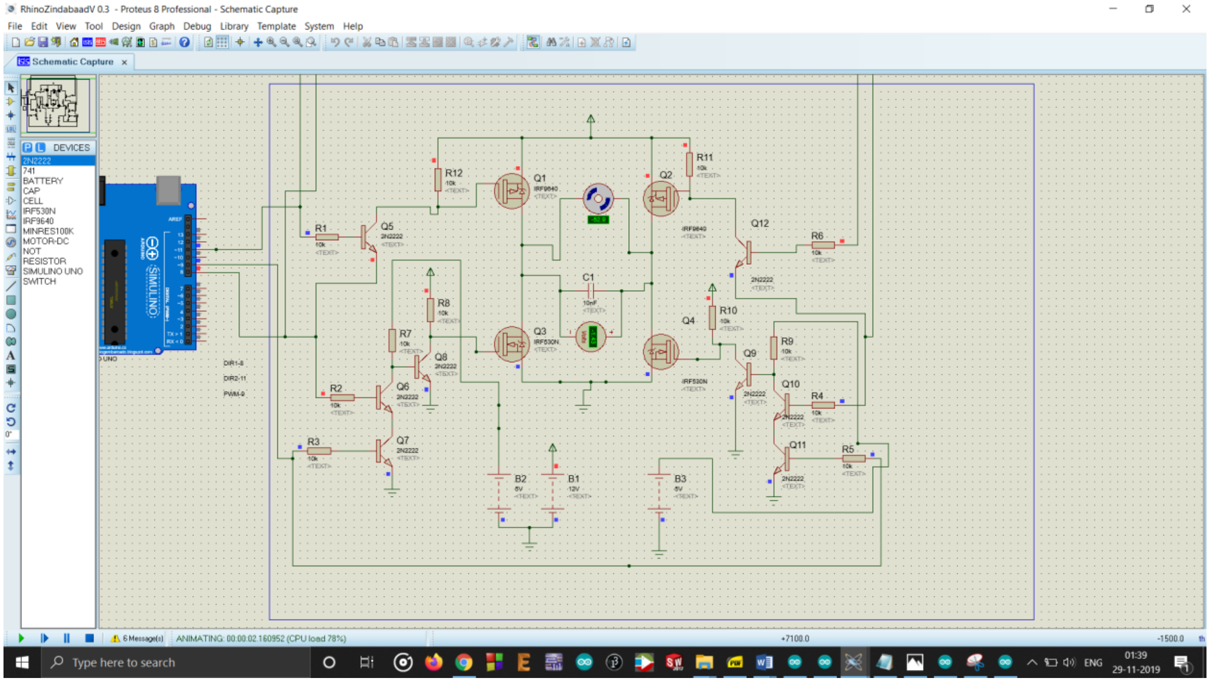

Fig No. 5.6 Simulation Diagram on Proteus

Before reaching to the making of the present motor driver we tried to simulate the above shown motor driver circuit. But it involved use of many transistors. So, we optimized the circuit with minimum usage of transistor and we replaced it by the signal MOSFET to drive high side switches, i.e., P-channel MOSFETs

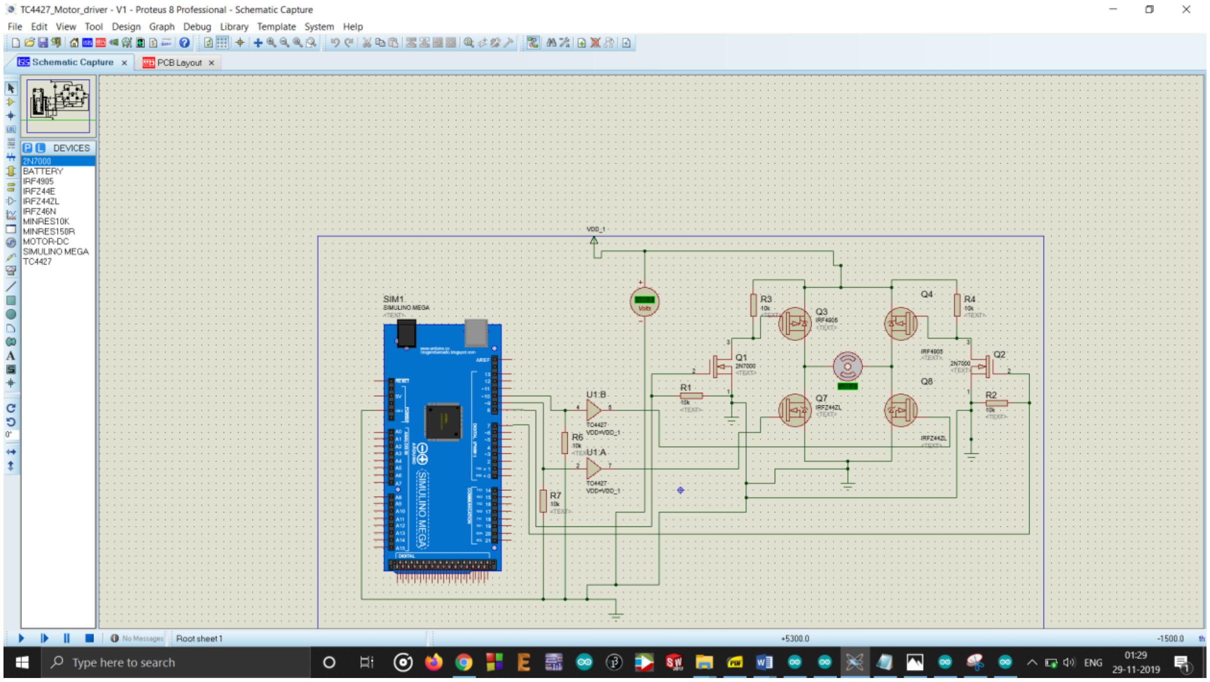

Fig No. 5.7 Schematic Diagram

5.3 Arduino Code :

```
  // defining pins to variables
int m1_AHS = 8;
int m1_ALS = 9;
int m1_BLS = 10;
int m1_BHS = 7;
void setup() {
  // pins 9 & 10 on standard Arduino or pins 11 and 12 on Arduino Mega
  TCCR1B = TCCR1B & 0b11111000 | 0x01;
  Serial.begin(9600);
pinMode(m1_AHS, OUTPUT);        // set motor pins as outputs
pinMode(m1_ALS, OUTPUT);
pinMode(m1_BHS, OUTPUT);
pinMode(m1_BLS, OUTPUT);
}
void loop() {
  // Command motor forward at speed 255
digitalWrite(m1_AHS, HIGH);
digitalWrite(m1_BLS, HIGH);
digitalWrite(m1_BHS, LOW);
analogWrite(m1_ALS, 255);
delay(1000);
}
```

6. TESTING :

6.1 Pre-Test :

- **Pin-outs :**

1. **Header Pin 1 (far left) :** Controls the AHI (A high-side input) of each bridge, which turns on the P-channel power MOSFET using the N-channel signal MOSFET. This input should be connected to the Gate pin of the 2n7000.

2. **Header Pin 2** : Connects to the TC4427 pin 2 (IN-A), which controls the ALI (A low-side input) of each bridge. You should use a PWM signal to control this input.

3. **Header Pin 3** : Connects to Ground.

4. **Header Pin 4** : Connects to the TC4427 pin 4 (IN-B), which controls the BLI (B low-side input) of each bridge. You should use a PWM signal to control this input.

5. **Header Pin 5 (far right) :** Controls the BHI (B high-side input) of each bridge, which turns on the P-channel power MOSFET using the N-channel signal MOSFET. This input should be connected to the Gate pin of the 2n7000.

Arduino variables	***Hardware connection***	***Arduino pin***
m1_AHS	N channel signal MOSFET	8
m1_ALS	Input A of TC4427	9
m1_BLS	Input B of TC4427	10
m1_BHS	N channel signal MOSFET	7

Fig No. 6.1 Pin-outs

- **Oscilloscope :**

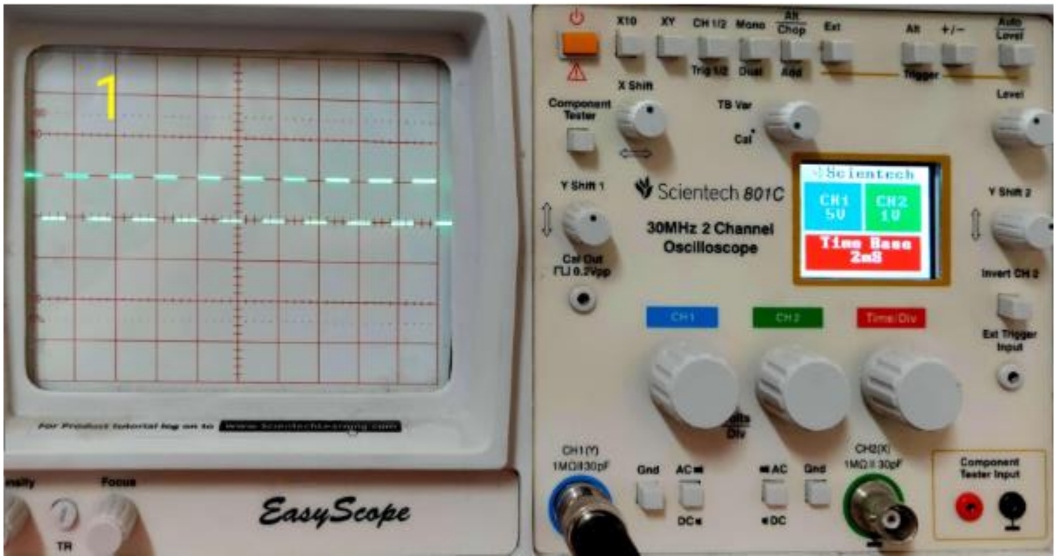

Fig No. 6.1 Test - 1

Arduino pin 9 connected to oscilloscope without change in frequency

T= 2 m sec / 5 division; T= 0.4 m sec / division; 6 division=6*0.4 m = 2.4 m sec

Frequency = 1/T = 416.67 Hz

According to datasheet of Atmega 328PU, the frequency at pin 9 is 420 Hz and we found the frequency to be 416.67 Hz

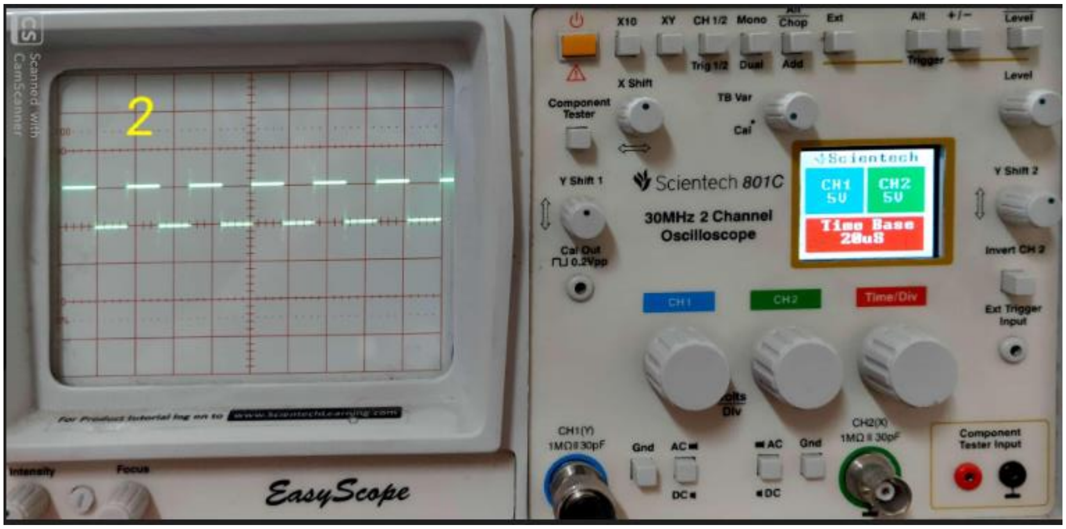

Fig No. 6.2 Test - 2

Arduino pin 9 connected to oscilloscope with change in frequency

T= 20 µ sec / 5 division; T= 4 µ sec / division; 8 division=8*4 µ = 32 µ sec

Frequency = 1/T = 31250 Hz

According to datasheet of Atmega 328PU, the frequency at pin 9 is 31250 Hz and we found the frequency to be 31250 Hz

Fig No. 6.3 Test - 3

Connection of TC4427 to Arduino without providing power to it will make no change in either frequency or voltage

Fig No. 6.4 Test - 4

Connection of TC4427 to Arduino with providing power of 8 V to will change the voltage from 5 V to 8 V but the frequency will remain as it was i.e., 31250 Hz

Fig No. 6.5 Test - 5

Connection of TC4427 to Arduino with providing power of 12 V to will change the voltage from 5 V to 12 V but the frequency will remain as it was i.e., 31250 Hz

So, from the above tests we observed that proving pwm signal to MOSFET driver will not change the frequency, only the voltage will change and the current is amplified.

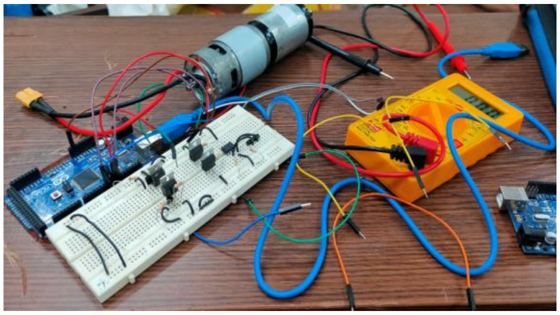

Fig No. 6.6 Implementation

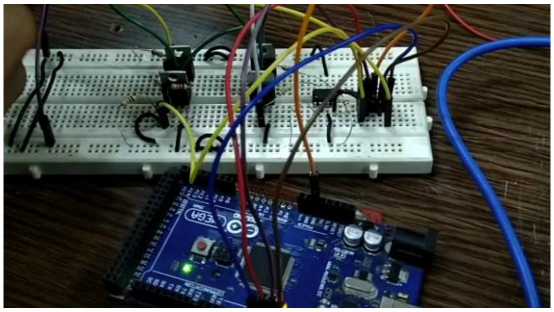

Fig No. 6.7 Apparatus

While doing the first test we initially did not used motor as load, but instead we used multimeter to check the output voltage. So, we observed the output voltage of approximately 4.5- 5V for 50% pwm when we connected battery of 12 V.

Fig No. 6.8 Multimeter Reading

The problem in this circuit was that it was implemented on bread board, so if there were any loose connections, it was difficult to debug.

At the end it was ended up with blowing of LOW side switch i.e., N channel MOSFET that was connected to the MOSFET driver was burnt due to high frequency switching, so performed the second test by replacing the N channel MOSFET and using zero PCB and soldering the components so that there are no loose connections.

6.2 Current Test :

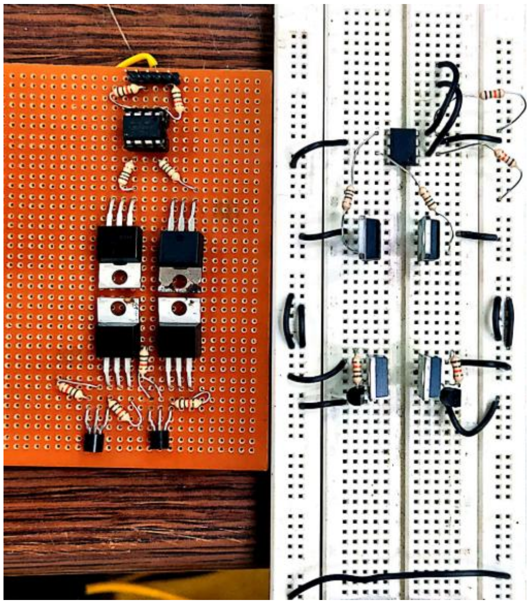

Fig No. 6.9 Pre-Test and Current Test Circuits

As in the test 1 the lower N-channel MOSFETs were burnt due to high frequency switching, it was replaced with another MOSFETs which easily bearded that much frequency without getting blowing up.

The second upgrade was the use of perforated zero PCB instead of zero PCB which ensured that there are no any loose connections and all the components are soldered firmly on the board.

This saved time in debugging the circuit.

When the load through the drain of the MOSFETs increases it draws more current which is not possible to implement it on the breadboard which can be used for the current lesser than 1 A. But when we use this zero PCB we can solder the metal according to our current rating.

But this circuit also ended up with short circuit, but there was no damage of any component when it was tested individually. Just the positive and negative terminals were shorted but it was switched off on time.

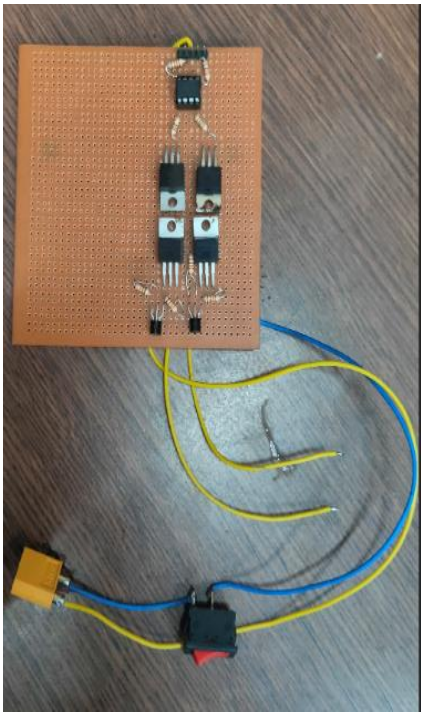

Fig No. 6.10 Before Short circuit

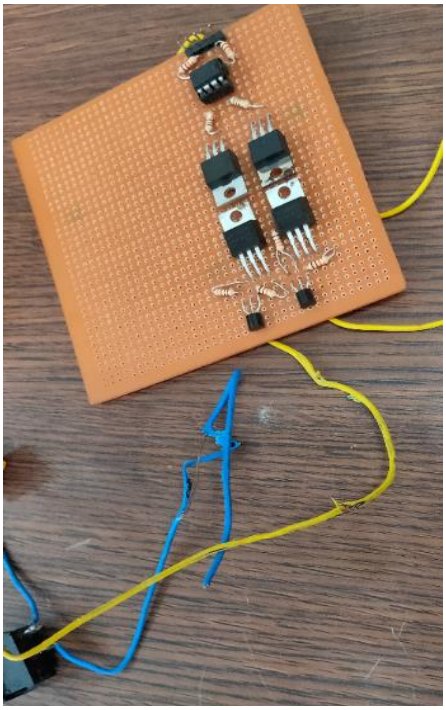

Fig No. 6.11 After Short circuit

6.3 Configuration Test :

In this test we again jumped back to bread board as it was becoming difficult to replace components again and again as we need to desolder the components in order to change them.

But as compare to the first test circuit this circuit was less complicated as the jumpers used were of limited required size so it was easy to debug the circuit.

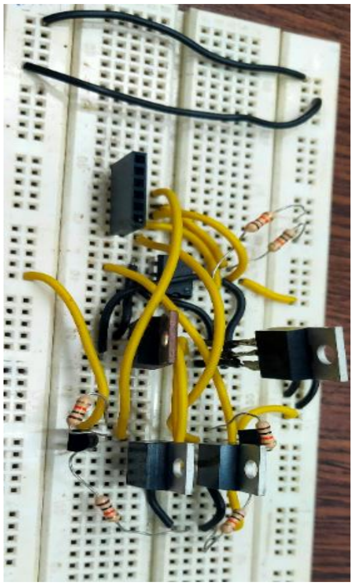

Fig No. 6.12 Implementation - 3 circuit

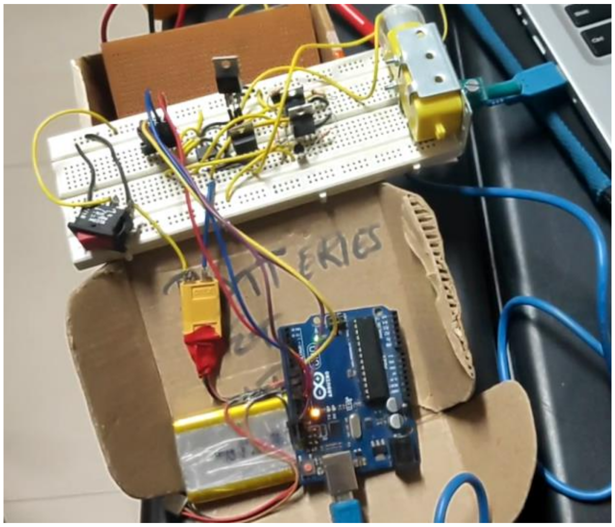

Fig No. 6.13 Implementation - 3 Apparatus

The complete circuitry was same just the space was optimized.

We connected a motor to the drain as a load. It successfully rotated in clockwise direction.

Now this was the final circuit and we decided to make a Printed Circuit Board for the same circuit and fix the components so that there are no any loose connections and it does not ends up with short circuit.

7. COST ANALYSIS :

Part	Description	Price per piece	Quantity	Cost
(2) N-channel signal transistors	(part #2N7000) —These are 200ma logic-level N-channel MOSFETs used to interface the P-channel power MOSFETs.	Rs. 14	2	Rs. 28
(2) N-channel power MOSFET	(part #FQP55N06) —Any N-channel power MOSFET will work. Rated for 52 amps at 60 volts.	Rs. 28	2	Rs. 56
(2) P-channel power MOSFET	(part #IRF4905) —This MOSFET has a high Gate-to-Source voltage (Vgs) of +/–25v, though any P-channel power MOSFET will work. Rated for 47 amps at 60 volts.	Rs. 90	2	Rs. 180
(1) TC4427 MOSFET drivers	(part #TC4427) —This MOSFET driver is used to switch the N channel power MOSFETs at high PWM speeds (32kHz) for silent operation.	Rs. 60	1	Rs. 60

(6) 10k resistors	This is used as a pull up resistor	Rs. 2	6	Rs. 12
(2) 150-ohm resistors	This is to limit the current from the MOSFET driver	Rs. 2	2	Rs. 4
(1) Female 6-pin headers	These headers are the same type that are used on the Arduino analog A0-A5 pins and make wiring the motor controller to the Arduino easy.	Rs. 20	1	Rs. 20
Perforated PCB prototyping board	This is used to mount the components and solder them together	Rs. 150	-	Rs. 150
			TOTAL	Rs. 510

Table No. 7.1 Bill of Material

8. FUTURE SCOPE :

The above made driver can be updated by building a simple MOSFET H-bridge that also has a current sensor built in to allow for adjustable current limiting. The current-limiting feature is important because it is the only way you can know if the motors are drawing too much current for the motor controller to handle. The result of drawing too much current through your motor controller is usually a burns PCB copper trace, or a blown up MOSFET! By monitoring the current-sensor IC with the Arduino, you can instantly stop either motor if it exceeds the preset level.

By increasing the rating of the MOSFETs, the driver of higher rating can be made. In addition to using higher-powered MOSFETs than the last motor controller, this board gets double MOSFETs. That is, by placing two MOSFETs in parallel on each leg of both H-bridges to double the current carrying capacity of the motor controller by halving the resistance.

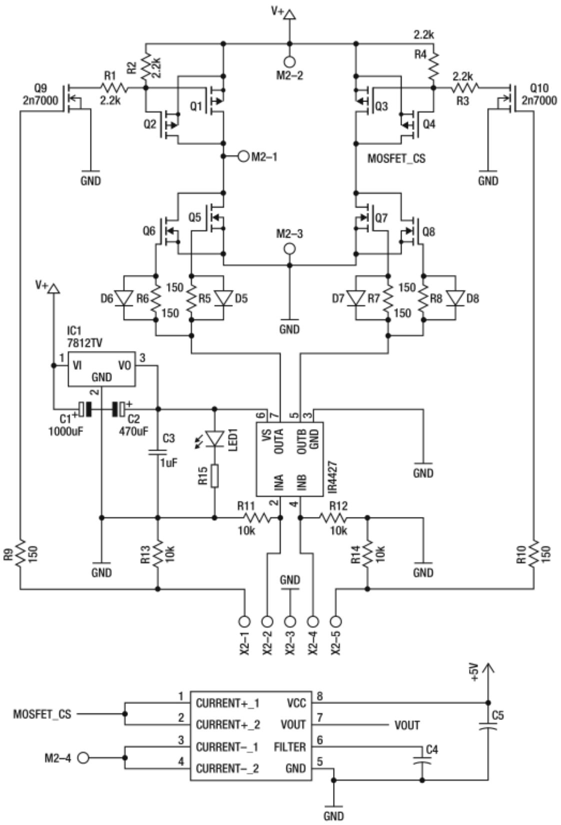

Fig No. 8.1 Future Idea

9. CONCLUSION :

There is a huge bag of knowledge gained at the end of this research work. And none of that would be possible without actually implementing it and verifying all the data about Motor Driver available over the internet or in books. The things we experienced about what goes wrong when what is applied, what all things give whichever results and what not to do while designing and implementing a Motor Driver. We noted a lot of points and the major one were :

- The amount of positive voltage applied to a Gate Driver (specifically TC4427) gives the output signal having frequency of it's input signal and amplitude of voltage provided
- Each and every Motor Driver requires a MOSFET Gate Driver to give fast switching frequency to the H-Bridge MOSFETs
- The P-channel MOSFETs in H-Bridge requires negative voltage to switch which is provided by another cascade N-channel MOSFET
- Whereas the N-channel MOSFET requires positive amplified input for switching provided by the MOSFET Gate Driver
- Individual Frequency of a I/O pin of Arduino can be customized through configuring Timer registers of the microcontroller
- We can easily change the current rating of the Motor Driver by getting higher current capable MOSFETs

REFERENCES

1. https://electronicsforu.com/resources/learn-electronics/mosfet-basics-working-applications
2. https://en.wikipedia.org/wiki/Frequency
3. https://learn.sparkfun.com/tutorials/pulse-width-modulation/all]
4. https://c03.apogee.net/mvc/home/hes/land/el?spc=foe&id=4619&utilityname=wppi
5. https://en.wikipedia.org/wiki/H_bridge
6. https://en.wikipedia.org/wiki/H_bridge#cite_note-williams_stamp-1
7. https://arduinoinfo.mywikis.net/wiki/Arduino-PWM-Frequency
8. https://www.instructables.com/id/MOTOR-DRIVER-30Amp/