23.01.2019

MIDDLE EAST TECHNICAL UNIVERSITY ELECTRICAL-ELECTRONICS ENGINEERING DEPARTMENT

EE463 POWER ELECTRONICS I

Fall 2018

HARDWARE PROJECT REPORT

Team Name: Smart Grid

Team Members: İven GÜZEL 2030831

Onur KÜLAHLIOĞLU

Ekin SAÇIN

To be submitted to: Ozan KEYSAN

Table of Contents

1. INTRODUCTION

This report presents the details of EE463 Hardware Project. The aim of this project is to design and complete the setup of a controlled AC to DC motor drive. The teams must define a topology in order to make a proper design for feeding motor from three phase AC supply.

All requirements and results for selected topology is presented in this report. Selected components for chosen topology are listed. The simulation results and design schematic are provided Simulink, MATLAB. Finally, the test results of the demonstration are presented due to using equipment in laboratory.

2. DESIGN DECISIONS

2.1. Topology Selection

In this project, we were required to implement a controlled rectifier for the armature voltage input of the motor. According to topologies we have learned this semester we had mainly three options.

3- Phase or Single-Phase Thyristor Rectifier:

For these solutions, we believe that 3 phase thyristor rectifier is preferable over the single phase one, since with a dc link capacitor connected to its output terminals, it offers a more dc like voltage waveform. In these topologies output voltage value is controlled with firing angles of thyristors. It should be noted that each thyristor requires its own driver and implementing gate driver circuits and the controller unit is a serious problem and we decided that it would be a challenge for our design. Also, synchronization problems could occur between firing angles and phases.

3 Phase Diode rectifier+ Buck Converter:

This solution requires a three-phase diode rectifier, a switch which could be MOSFET or IGBT, a diode, a DC link capacitor, gate driver circuit for the switch and a controller unit. Output voltage is controlled with Buck converter. For Buck converter a low pass filter could be used, however this part can be skipped since the motor itself is an LR low pass filter.

This solution requires only a single gate driver circuit; therefore, it possesses no synchronization problem like in the thyristor case. We believe this is the simplest solution; therefore, we decided to go with it.

3 Phase Diode + Boost Converter:

Although this solution is very similar to the previous one, we decided not to use it since boost converter increases the voltage however it reduces the current input.

2.2. Gate Driver

An isolation is needed between the microcontroller of the gate driver and the controlled rectifier topology. For this purpose, we used an optocoupler (TLP250) as our gate driver circuit as it can be seen in Figure 2.1. From low voltage side of the optocoupler we implement the duty cycle information (PWM) using an Arduino Uno.

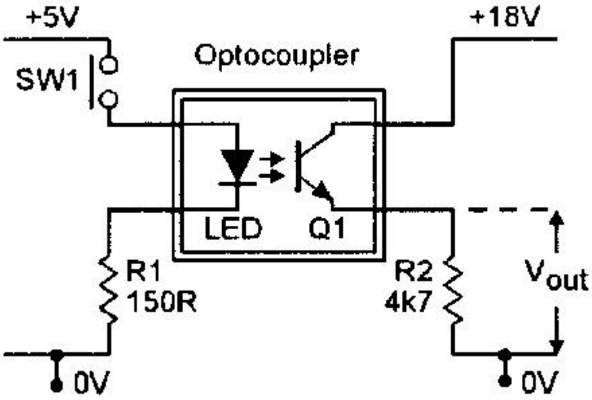


Figure 2.1- Optocoupler as the gate driver

Low voltage side of the optocoupler can be thought as a LED. PWM output of Arduino is conected to anode of the LED. A resistor is needed to be connected to the cathode of the LED in order to prevent it from burning. A floating voltage source should be connected between Vcc and Vee legs of the optocoupler, since to open the gate of the IGBT or Mosfet that are used as switch, the voltage at the gate should be bigger that the summation of the threshold voltage of the switch and emitter voltage (for IGBT) or source voltage (for Mosfet). Therefore; the lower port of this floating source is connected to the emitter of the IGBT (or source of the Mosfet). To feed the gate any one of two middle legs of the Optocoupler could be used.

We create the PWM input using Arduino. However, switching of the IGBT should be in high frequencies and therefore; a manipulation is needed to done on Arduino's clock. The source code given in Appendix A, increases switching frequency to 3.6kHz and slowly increases duty cycle from 0 to 0.94 and decreses it back to 0 again.

3. COMPUTER SIMULATIONS

After choosing the topology of the driver, we started to determine the proper equipment by analyzing the simulation results. Voltage ripple, average output voltage, and current were adjusted by changing the parameters of the equipment. We measured the terminals of the DC link capacitor to determine the voltage ripple by changing the capacitor values and measure the terminals of the diode as output voltage. Also, we measured the armature current of the DC motor. We tried to adjust output voltage at 220 Volts which is the rated voltage of the machine. We used 185 V, rms voltage as input voltage of the driver circuit.

In this part, some simulations were made by MATLAB Simulink for three different cases. We arranged the parameters of the equipment with respect to the datasheet of the equipment that we bought. In these cases, the duty cycle was changed and some important parameters were observed. The value of duty cycle was adjusted with the pulse generator. The circuit schematic of simulations is in Figure 3-1.

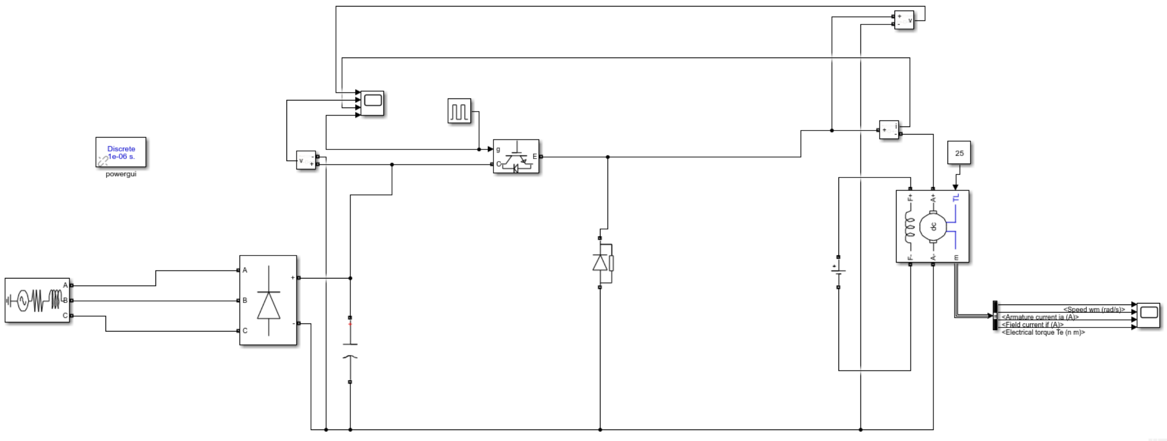
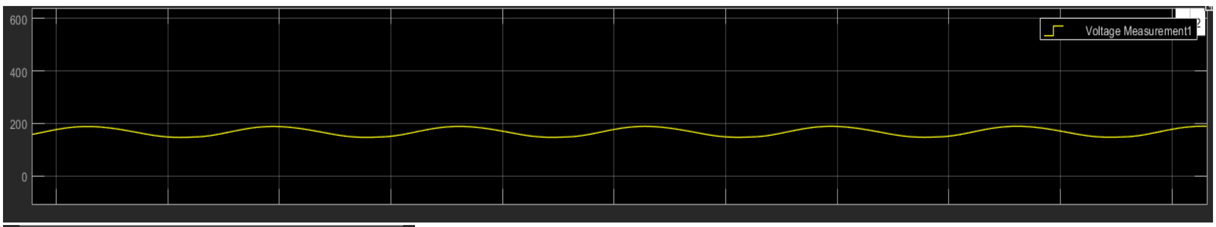


Figure 3.1- The circuit schematic of converter in Simulink

As an extreme case, when we adjust the duty cycle of the gate-input as 1, we observed the following results seen in the Figure-3.2, 3.3 and 3.4.

- PWM Duty Cycle = 1



Signal Statistics

	Value	Time
Max	1.894e+02	0.281
Min	1.471e+02	0.262
Peak to Peak	4.232e+01	
Mean	1.680e+02	
Median	1.682e+02	
RMS	1.686e+02	

Figure 3.2- Diode Voltage

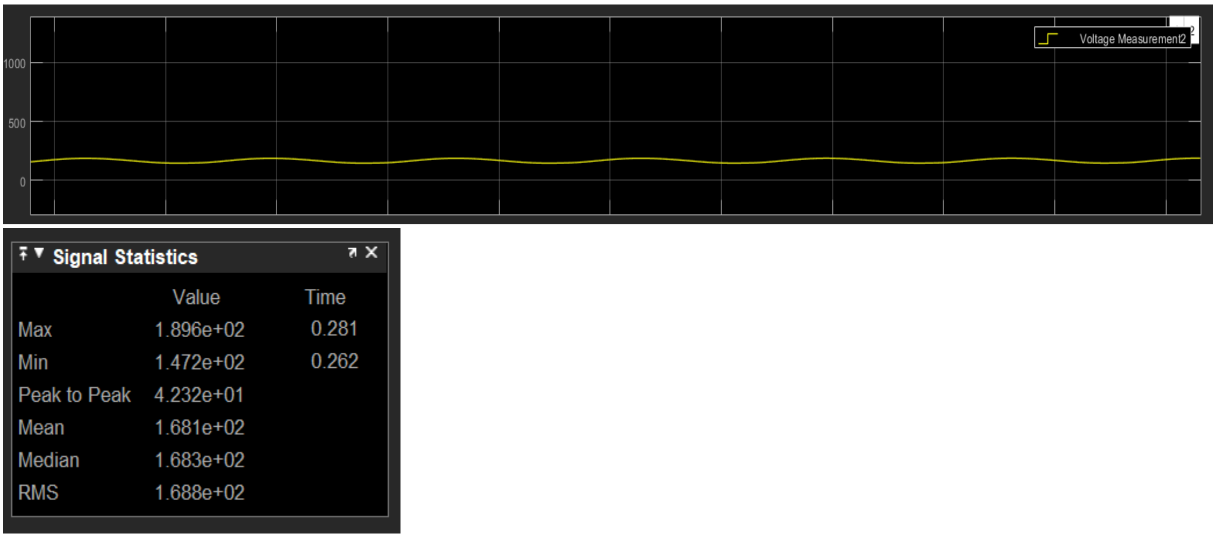


Figure 3.3- Voltage of DC link capacitor

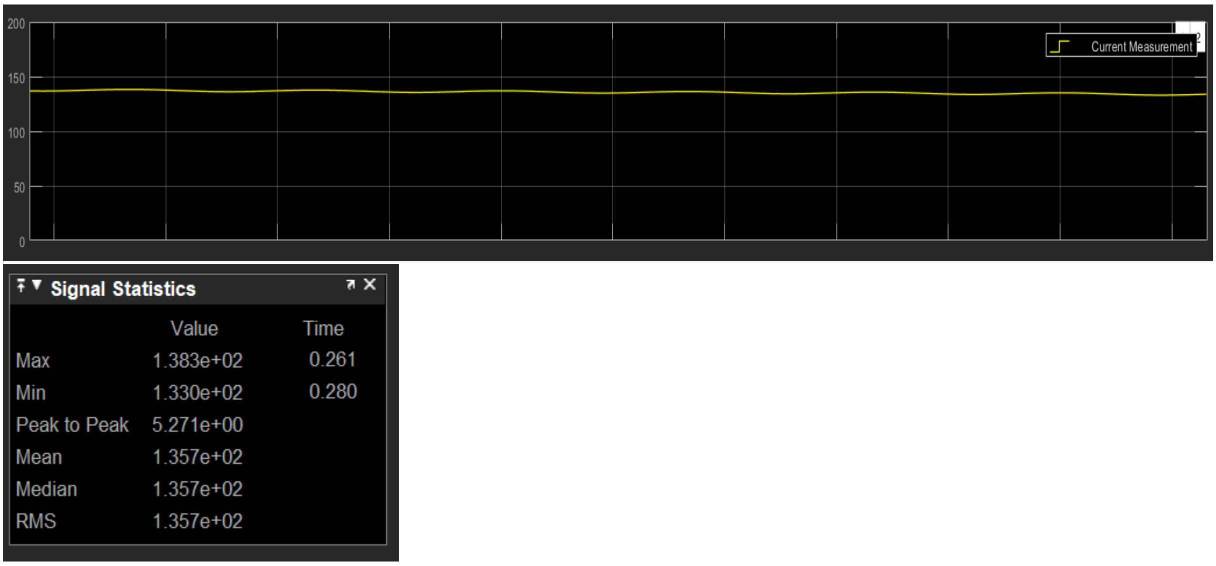
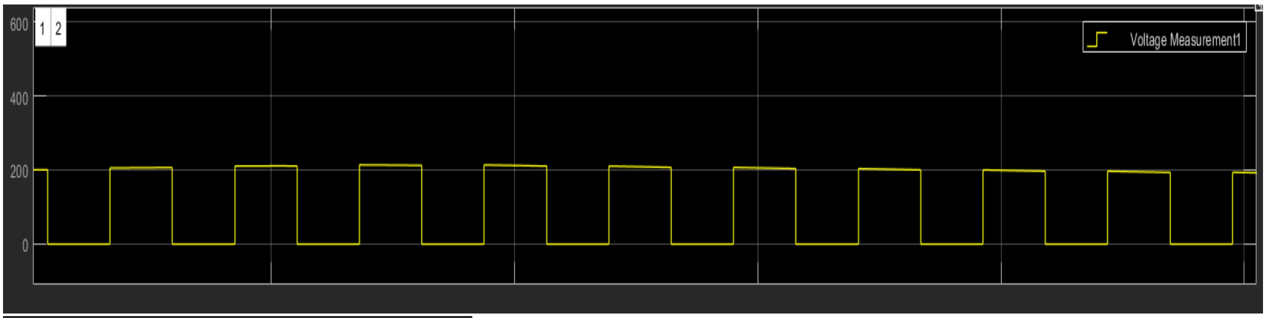
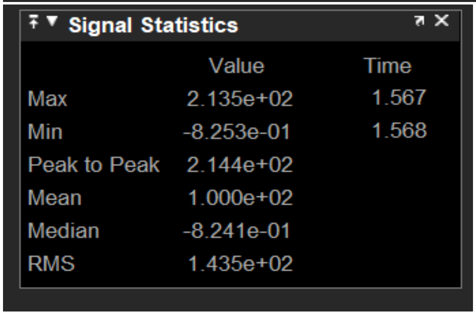


Figure 3.4- Motor current

To get more realistic results, we arranged the duty cycle of the gate-input as 0.5 and we observed the voltage of the DC link capacitor and diode as an input and output voltages as seen in the Figure-3.5 and 3.6. Besides, we obtained the armatüre current of the DC motor.

- PWM Duty Cycle = 0.5

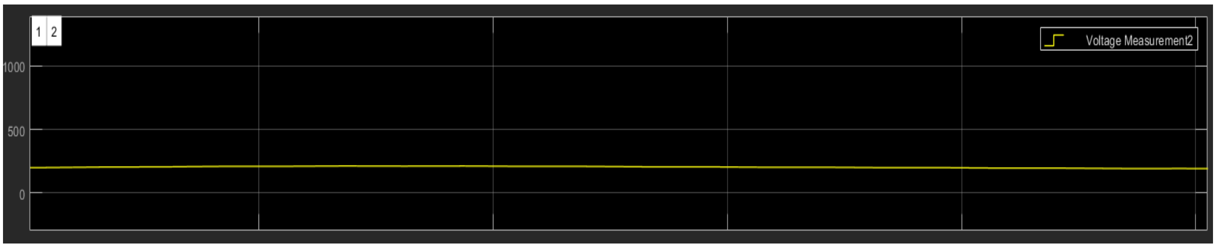


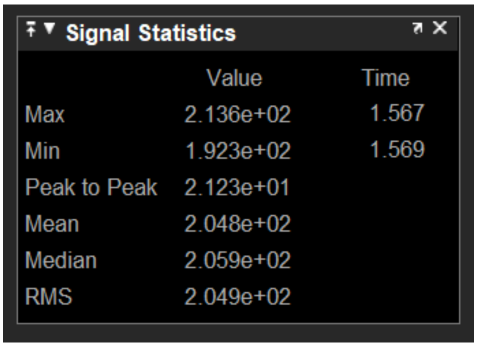


Signal Statistics

	Value	Time
Max	2.135e+02	1.567
Min	-8.253e-01	1.568
Peak to Peak	2.144e+02	
Mean	1.000e+02	
Median	-8.241e-01	
RMS	1.435e+02	

Figure 3.5- Diode Voltage

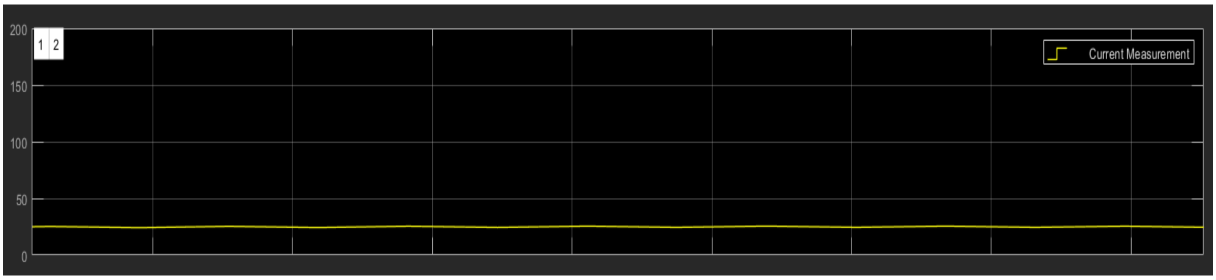


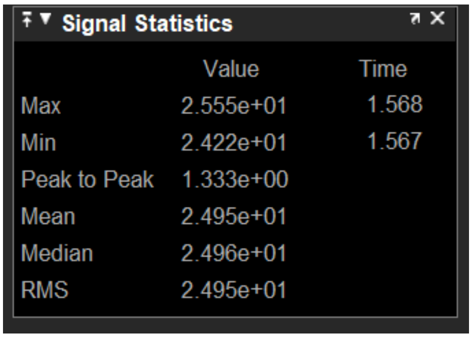


Signal Statistics

	Value	Time
Max	2.136e+02	1.567
Min	1.923e+02	1.569
Peak to Peak	2.123e+01	
Mean	2.048e+02	
Median	2.059e+02	
RMS	2.049e+02	

Figure 3.6- Voltage of DC link capacitor





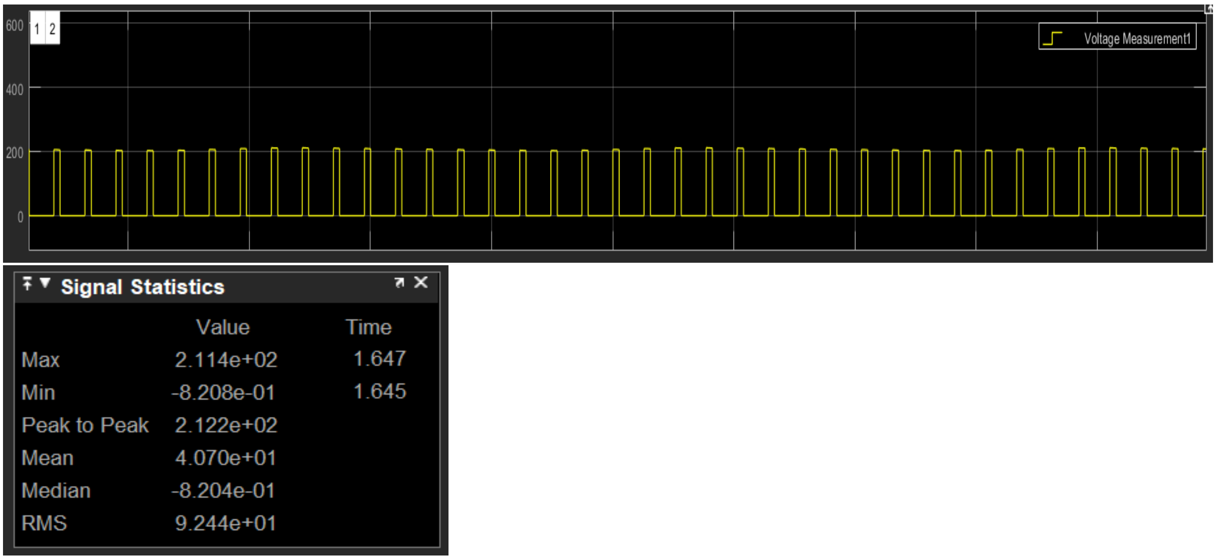
Signal Statistics

	Value	Time
Max	2.555e+01	1.568
Min	2.422e+01	1.567
Peak to Peak	1.333e+00	
Mean	2.495e+01	
Median	2.496e+01	
RMS	2.495e+01	

Figure 3.7- Motor current

As another extreme case, we changed the duty cycle of the gate-input as 0.2. Probably, it is not enough to rotate the machine, but we wanted to observe the characteristics of the circuit at this duty cycle. The simulation results can be seen in the Figure- 3.8, 3.9 and 3.10.

- PWM Duty Cycle = 0.2



Signal Statistics

	Value	Time
Max	2.114e+02	1.647
Min	-8.208e-01	1.645
Peak to Peak	2.122e+02	
Mean	4.070e+01	
Median	-8.204e-01	
RMS	9.244e+01	

Figure 3.8- Diode Voltage

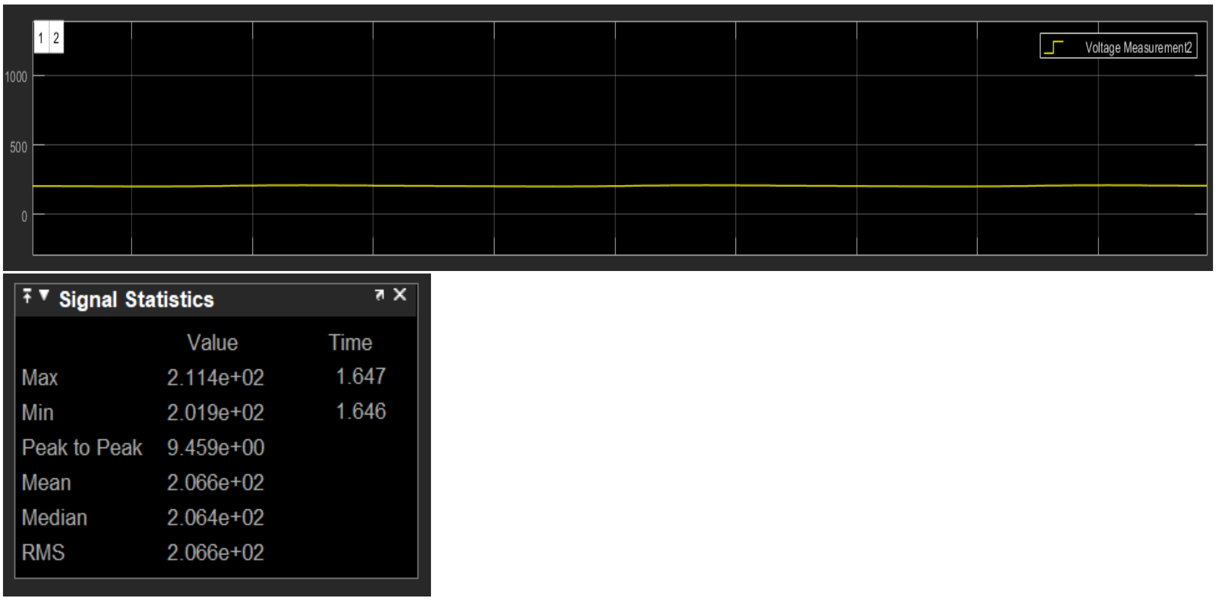


Figure 3.9- Voltage of DC link capacitor

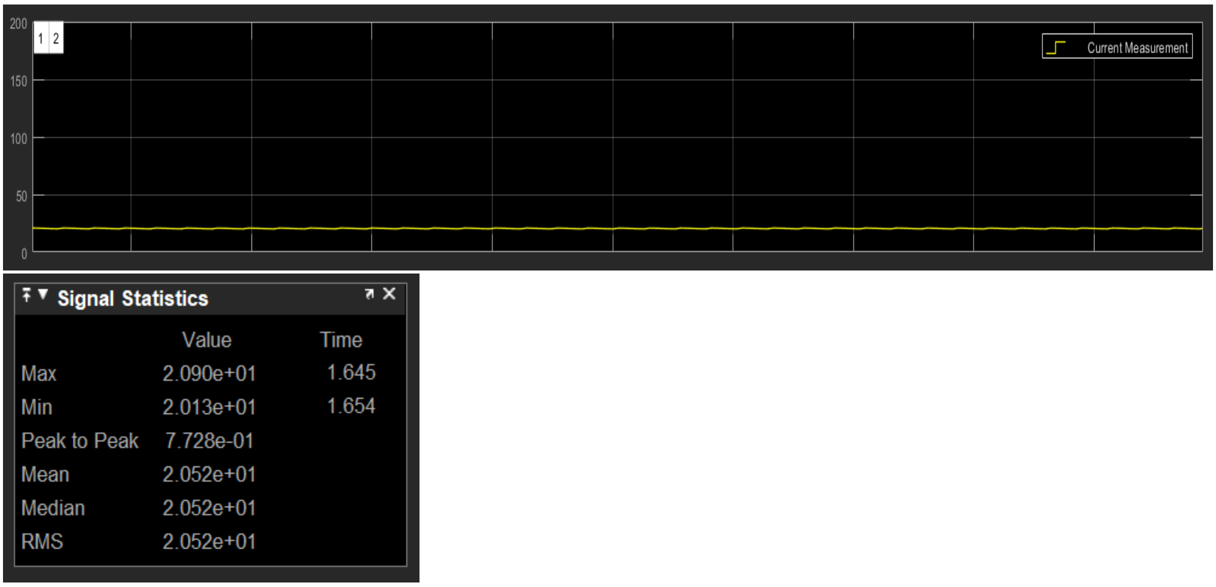


Figure 3.10- Motor current

After these simulations we obtained largest values when duty cycle is 1. Therefore; we conducted a final simulation with 0.95 duty cycle. Results of this simulation can be seen in Table 3.1.

Table 3.1: Simulation results for duty cycle = 0.95

IGBT Voltage RMS (on)	1.3
IGBT Current RMS (on)	6
Diode Voltage RMS (off)	-220
Diode Current RMS	6
Capacitor Voltage RMS	220
3 Phase Rectifier Voltage RMS	220

4. COMPONENT SELECTION

4.1. Equipment Selection

After conducting simulations in Simulink and obtained rated values for components, at first, we decided to buy components online therefore, we checked some datasheets. However, delivery time was too late. So, we decided to gather our equipment from Konya Sokak. Datasheets of components can be found in references section.

3-Phase Diode Rectifier:

We decided to use a 3-phase rectifier unit instead of using 6 separate diodes, since integration would be troublesome. We chose VUO25-14NO8 Standard Rectifier Module [1] as our rectifier. The rectifier can be seen in Figure 4.1. If we observe parameters listed below, peak reverse voltage breakdown is high for the project but average rectified output current capacity is considered when this component is selected.

- Repetitive Peak Reverse Voltage: 1400V
- Average Rectified Output Current: 20A
- Junction Temperature: -40-150 C
- Max. forward surge current: 380A
- Forward voltage drop: 1.05V
- Thermal resistance (junction to case plus case to heatsink): 9K/W

Figure 4.1 .Three phase diode rectifier that is used in the project

IGBT:

As the switch in our system IXSH 35N120A High speed, High Voltage IGBT is used. IGBT is a little more expensive than MOSFET but it is more reliable with respect to in rush currents. In Figure 4.2 our IGBT and in Figure 4.3, important parameters that we used to select this component can be seen. We chose this component considering current values firstly.

Figure 4.2 .IGBT that is used in the project

$t_{d(on)}$	**Inductive load, $T_J = 25°C$**	80		ns
t_{ri}	$I_C = I_{C90}$, $V_{GE} = 15$ V, $L = 100\ \mu H$; $V_{CE} = 0.8\ V_{CES}$, $R_G = 2.7\ \Omega$	150		ns
$t_{d(off)}$	Remarks: Switching times may	400	900	ns
t_{fi}	increase for V_{CE} (Clamp) > 0.8 • V_{CES},	500	700	ns
E_{off}	higher T_J or increased R_G	10		mJ
$t_{d(on)}$	**Inductive load, $T_J = 125°C$**	80		ns
t_{ri}	$I_C = I_{C90}$, $V_{GE} = 15$ V, $L = 100\ \mu H$	150		ns
E_{on}	$V_{CE} = 0.8\ V_{CES}$, $R_G = 2.7\ \Omega$	2.5		mJ
$t_{d(off)}$	Remarks: Switching times may	400		ns
t_{fi}	increase for V_{CE} (Clamp) > 0.8 • V_{CES},	700		ns
E_{off}	higher T_J or increased R_G	15		mJ
R_{thJC}			0.42	K/W
R_{thCK}		0.25		K/W

$$V_{CES} = 1200\ \text{V}$$
$$I_{C25} = 70\ \text{A}$$
$$V_{CE(sat)} = 4\ \text{V}$$

Figure 4.3 : Ratings of IGBT

<u>Freewheeling diode:</u>

We used DSEI60-06A Fast Recovery Epitaxial Diode (FRED) as the freewheeling diode in the Buck converter in Figure 4.4.

Figure 4.4 .Photo of the diode that is used in the project

While selecting this diode switching characteristics, current/voltage ratings are considered. Since we switch the IGBT with 3.9kHz diode must be fast enough. During off stage of IGBT, motor current circulates through the diode which means current/voltage ratings of diode must be equal or higher than the motor ratings. As it can be seen in Figure 4.5, this component satisfies our needs.

$$I_{FAV} = 60 \text{ A}$$
$$V_{RRM} = 600 \text{ V}$$
$$t_{rr} = 35 \text{ ms}$$

Figure 4.3 : Ratings of free-wheeling diode

DC Link Capacitor:

A capacitor that is 270uF that can resist 450V continuous voltage is used. However, we do not have a datasheet for this component.

Optocoupler:

We used TLP250 [4], as an isolator between Arduino and gate driver circuit. How to use this component is explained in Gate Driver section.

Arduino:

Arduino Uno is used as microcontroller to generate PWM signal. How to use this component is also explained in Gate Driver section.

4.2 Cost Analysis

MAIN COMPONENTS	COST(TL)
3-Phase diode rectifier module	60
IGBT	40
Freewheeling diode	20
Capacitor(x3)	3x20
Arduino	23
Optocoupler	15
Heat Sink(x3)	3x5
Ventilator	20
Stripboard	10
Plastic box	40

Other than these components, we have used ceramic capacitors, resistors, cables, connectors, thermal paste, sockets and battery.

5. TEST RESULTS

We have tested all subparts after purchasing them to make sure that they are suitable for the project. Firstly, we have tested diode rectifier module and 540 µF DC link capacitor by using resistive load. As seen in the Figure-6.1, output voltage has almost no ripple thanks to the capacitor.

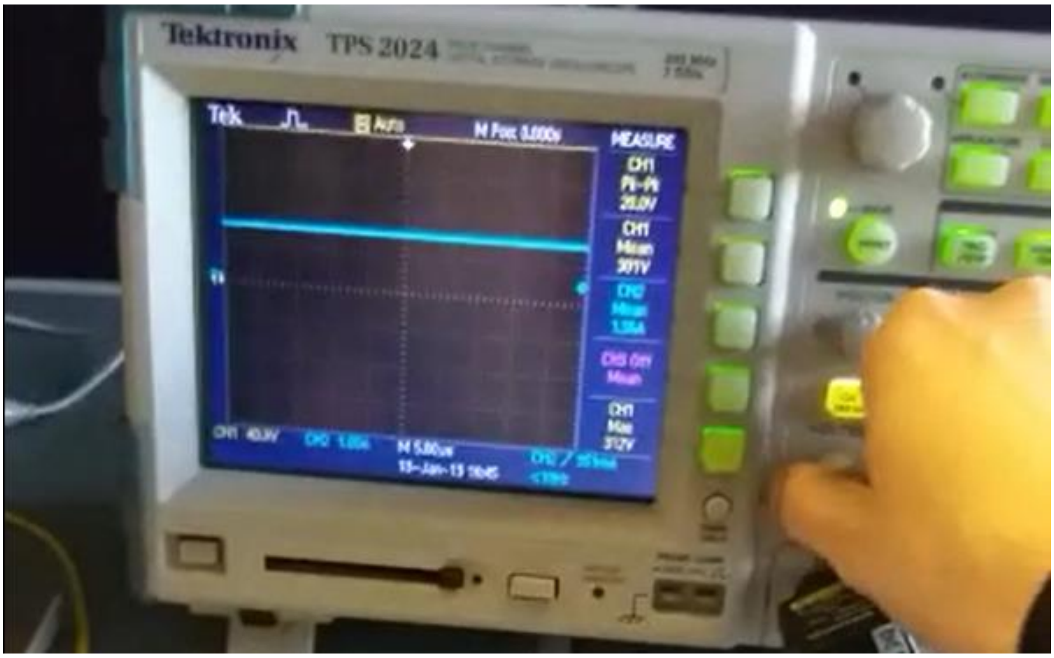


Figure-6.1: Output voltage of the diode rectifier module with DC link capacitor

We also looked at the gate signals of the IGBT generated by the Arduino with the help of an oscilloscope. Results for 2 different duty cycle is seen in the Figure-6.2.

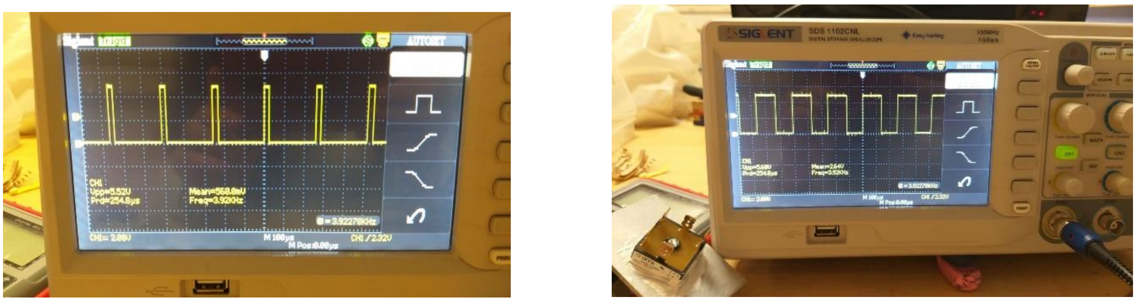


Figure-6.2: Output of the Arduino with 2 different duty cycle levels.

After tested the 2 subsystems mentioned above, we build the whole circuit and tested with resistive load and low input voltage. We observed overvoltage at the beginning of each cycle of the output voltage. This is dangerous for high power application. This could have burned our components, so we decided to use RC snubber to decrease the overvoltage. You can see the effect of the snubber in the Figure-6.3.

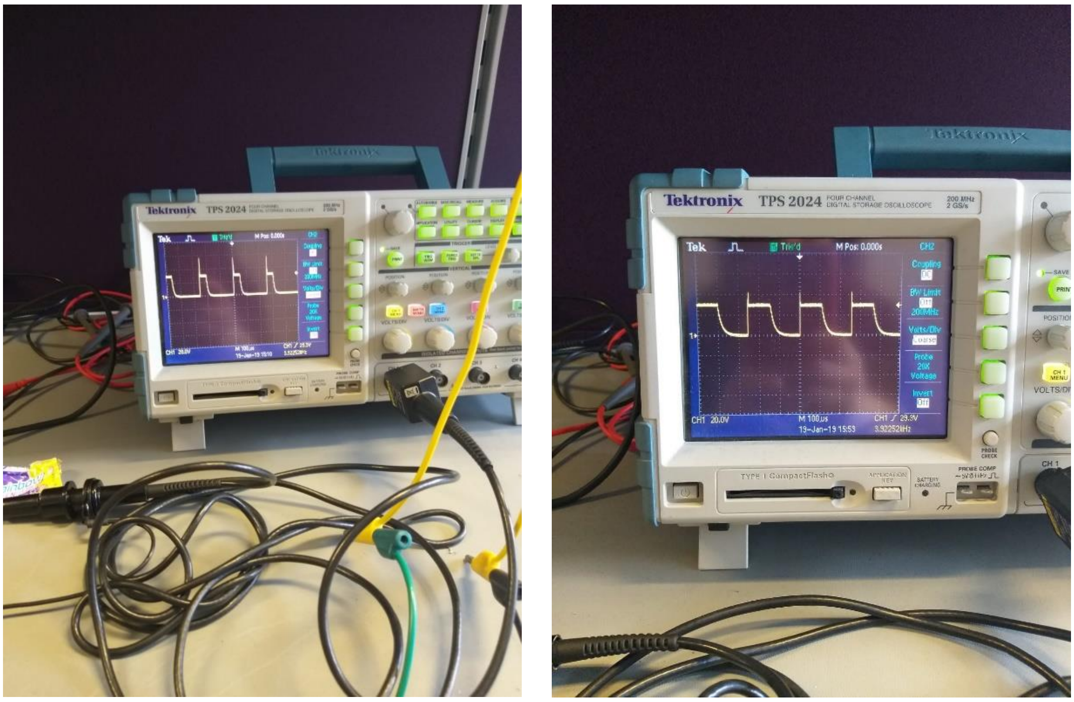


Figure-6.3: Output voltage without snubber (left side) and with snubber (right side).

Finally, we were ready to test our circuit with RL load. Test results were what we expected as seen in the Figure-6.4. We managed to control the output voltage of the load by changing the duty cycle of the gate voltage of the IGBT.

Figure-6.4: Output voltage and current with RL load with different duty cycles

All tested were successful at the power electronics laboratory, so we decided to test our design in the machinery laboratory with DC motor. Our driver was operating well until there were some problems with the DC motor. Results before the problem are seen in the Figure-6.5.

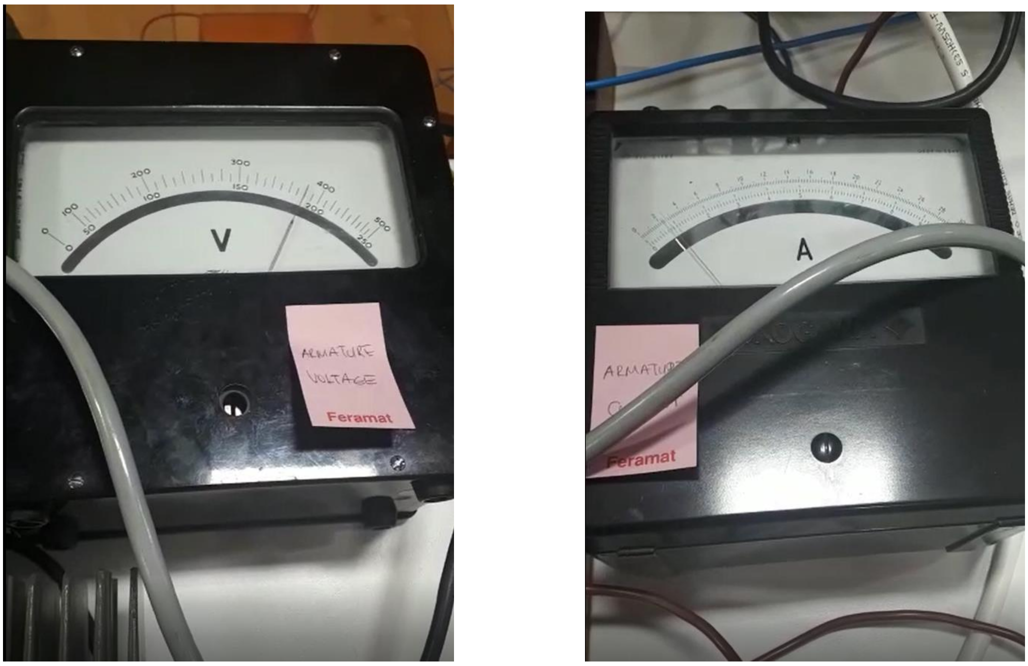


Figure-6.5: Input voltage and current of the DC motor.

We tried to arrange the output voltage of the driver at the rated voltage of the DC motor which is 220 Volts. Motor was drawn about 2.2 amperes at no load and 6 amperes at full-load. At the full-load, motor was drawn about 10 amperes on the demo day due to effect of series resistance at the load side. At no load, the temperature of the driver was around 30 degrees as seen in the figure-6.6, but at the full load, it was up to 70 degrees.

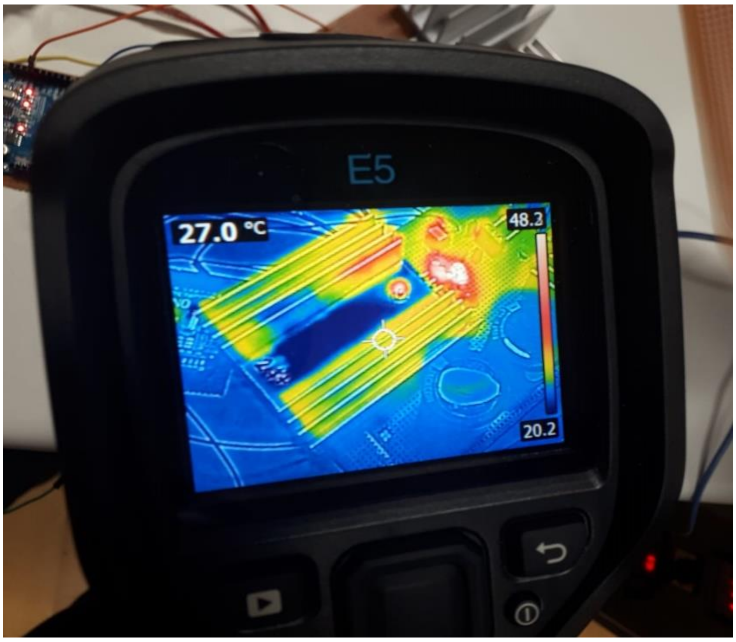


Figure-6.6: Input voltage and current of the DC motor

6. CONCLUSION

In this project, the target is to design and implement a DC motor controller which should deliver minimum 2 kW power during at least 5 minutes. For these requirements, we decided to use Buck Converter topology in our design. This topology divides into two parts. In the first part three phase AC converting to DC and in the second part, there is DC-DC step down converter.

Motor speed controlling is also expected from us while covering the design. For this purpose, we used Arduino board to control Buck converter, and it provides PWM output due to speed demand.

During covering our design, we have made a lot of simulations on Simulink, MATLAB, and gathered data for the selected topology. Moreover, these results help us while selecting components used in our design.

During the implementation, we faced with a lot of problems. One of the most familiar problem is that IGBT is burned out because of large amount current due to the instant of sharp input set point changes of the motor. We placed the snubber to gate of IGBT not to face with this problem. Snubber works by limiting instant sharps, but even though our IGBT is burned out since we started to run motor with so high voltage from three phase supply.

To sum up, we got a lot of practical experiences which are different from lecture studies thanks to this instructive project. We learned the converter topologies, useful information about protection, manufacturability and managing budget. As a result, this project helps us to encourage our engineering skills.

We also recorded our experiences as video during working process as seen in the following link.

https://www.youtube.com/watch?v=s0M1ChqQ4oM&feature=youtu.be

APPENDIX A: Source Code of Arduino Gate Driver

```
int POT = A1;
int PWM =3;
float Duty=12;

float Switch =0;


void setup() {
  Serial.begin(9600);
  pinMode(PWM,OUTPUT);
  pinMode(POT,INPUT);
  setPwmFrequency(3, 8); //Sets PWM freq to 3.9kHz
}

void setPwmFrequency(int pin, int divisor) {
  byte mode;
  if(pin == 5 || pin == 6 || pin == 9 || pin == 10) {
    switch(divisor) {
      case 1: mode = 0x01; break;
      case 8: mode = 0x02; break;
      case 64: mode = 0x03; break;
      case 256: mode = 0x04; break;
      case 1024: mode = 0x05; break;
      default: return;
    }
    if(pin == 5 || pin == 6) {
      TCCR0B = TCCR0B & 0b11111000 | mode;
    } else {
      TCCR1B = TCCR1B & 0b11111000 | mode;
```

```
    default: return;
  }
  if(pin == 5 || pin == 6) {
    TCCR0B = TCCR0B & 0b11111000 | mode;
  } else {
    TCCR1B = TCCR1B & 0b11111000 | mode;
  }
} else if(pin == 3 || pin == 11) {
  switch(divisor) {
    case 1: mode = 0x01; break;
    case 8: mode = 0x02; break;
    case 32: mode = 0x03; break;
    case 64: mode = 0x04; break;
    case 128: mode = 0x05; break;
    case 256: mode = 0x06; break;
    case 1024: mode = 0x07; break;
    default: return;
  }
  TCCR2B = TCCR2B & 0b11111000 | mode;
}


roid loop() {
              Switch = analogRead(POT);
              Serial.print("Switch is:");
              Serial.println(Switch) ;


                if ((Duty<20)&&(Switch>400)){
```

```
void loop() {
                Switch = analogRead(POT);
                Serial.print("Switch is:");
                Serial.println(Switch) ;


                  if ((Duty<20)&&(Switch>400)){

                    while(Duty<240){
                      Duty = Duty +1;
                      analogWrite(PWM,Duty);
                      delay(100);
                      Serial.print("The duty cycle is:");
                      Serial.println(Duty) ;
                    }
                    }

               if ((Duty>230)&&(Switch<400)){

                    while(Duty>10){
                      Duty = Duty - 1;
                      analogWrite(PWM,Duty);
                      delay(100);
                      Serial.print("The duty cycle is:");
                      Serial.println(Duty) ;
                    }
                    }
```

REFERENCES

[1] http://ixapps.ixys.com/Datasheet/VUO25-14NO8.pdf

[2] http://ixapps.ixys.com/datasheet/92774.pdf

[3] http://www.farnell.com/datasheets/123255.pdf

[4] https://web.itu.edu.tr/yildiri1/mylibrary/data/tlp250.pdf