

REMOTE BATTERY STATE MONITORING IoT DEVICE FOR LIGHT ELECTRIC VEHICLE WITH BATTERY SWAPPING MECHANISM

Group: G26

Supervised By
Dr B.G.L.T Samaranayake

Group Members

R.A.O.S RANAWEEERA (E/14/279)
H.B.N.D.GUNATHILAKE (E/14/112)



**Department of Electrical and Electronic Engineering
Faculty of Engineering
University of Peradeniya
Peradeniya 20400
Sri Lanka
July 2020**

ABSTRACT

Remote IoT platforms becoming more and more popular in the engineering industry as to gather data from remote sources, make meaningful decisions based on the gathered data such as management provisioning and automation of services associated with the connected devices.

Our target of this project was to design and implement a complete eco system for the electrical vehicles, where the electrical vehicle owners can swap fully charged batteries via visiting a battery swapping station, once the battery state of charge is been depleted.

This project involved working with areas such as prototype mechanical design, hardware implementation, implementation of the IoT platform to work with mobile devices, design of a optimization algorithm to maximize the profits earned through the battery swapping station.

The main outcomes of the project could be listed as,

- CAD based 3d design of battery swapping mechanism, with multiple prototypes, using SOLIDWORKS 3D software package and verify the practical implement ability of the prototypes. This part also included market research for the various parts needs in order to implement this project and preparing of the BOQ.
- Practical implementation of the hardware prototype is done with the help of engineering workshop, and the materials for the prototype were obtained from the local market, online stores. The practical implementation also involved finding substitute materials for the parts which were not obtainable due to unavailability.
- Implementation of extended Kalman filter based battery state estimator on an android based mobile platform. This algorithm is initially implemented on a raspberry pi computer due to need of processing power for the calculations. In this approach the raspberry pi was replaced with a simple circuit that transfers the voltage and current readings from the battery to the customer mobile device via Bluetooth. And the State of Charge, State of Health and temperature of the battery is calculated in the customer mobile application
- Design of mobile interface for the end users that connects to the IoT platform and updates the users about the geographical locations of the battery swapping stations and state of charge and ability to reserve a battery in the nearby battery swapping station.
- Design of battery demand prediction model by using hourly data obtained from the IoT platform, two different approaches to estimate the future demand was implemented, the generated demand prediction was used for the optimization of the battery swapping station.
- In the first approach for the demand prediction, the time series data obtained from the IoT platform was used to fit the data into a statistical distribution, from which the demand was estimated.

- In the next approach for the demand estimation the ARIMA based modelling of the time series data was used to predict the demand for the next 24 hours.
- An energy source optimization algorithm was developed to maximize the profit earned through the battery swapping station. The profit maximization is done via selling the additional battery charge to the local grid supply. The optimization is performed via gradient decent and Levenberg Marquardt algorithm.
- Replacing the converter setup and superconductors with a Buck Boost converter to meet the required power limits.

ACKNOWLEDGEMENTS

First, we would like to express our sincere gratitude to our project supervisor Dr.B.G.L.T Samaranayake, for the guidance given throughout this course EE405 - Undergraduate Project I .We greatly appreciate the guidance given by him regarding this project ,providing us with the information related to this project and for the constant supervision throughout this semester.

Then we would also wish to thank Dr Asanga Rathnaweera from mechanical engineering department for the guidance given to us on the material selection, for the swapper prototype. Moreover we are grateful for the Dr.M.P.B.Ekanayake for the support given to us on the validation of the mathematical models we have used on this project.

Moreover, we are immensely grateful to the the staff at the engineering workshop, who worked energetically and tenderly towards the completion of the swapper prototype.We would like to specially thank Mr. Egodawatthe of the engineering workshop for the coordination of the workshop human resources and other machinery for the construction of the swapper prototype.And we would also like to thank Mr Indrajith ,Mr Jayathilake ,Mr Mel from the engineering workshop for their contribution for the completion of the hardware prototype.

Furthermore we would like to thank Mr Anura, from the mechanical engineering department who give insights and suggestions on the designed prototypes of our project.We would also like to specially thank E.M. Sachini Piyoni Ekanayake , for her prior work associated with this project and guiding us with the work already done by her on this project.

We would also thank course coordinator of EE405, Prof. J.B. Ekanayake, for the encouragement given to us by organizing lecture series on how to proceed with this project in an effective way and also for structuring the course in an effective manner.

Finally, we wish to extend our sincere gratitude to the lecturers, fellow colleagues and all other individuals who gave their ideas, support and encouragement that empowered us to make our project a success.

R.A.O.S RANAWERA (E/14/279)
H.B.N.D.GUNATHLAKE (E/14/112)

Department of Electrical and Electronic Engineering,
Faculty of Engineering,
University of Peradeniya.

TABLE OF CONTENTS

ABSTRACT	i
ACKNOWLEDGEMENTS	iii
LIST OF FIGURES	vi
LIST OF TABLES	ixx
LIST OF ABBREVIATIONS	x
CHAPTER 1 INTRODUCTION	1
1.1 OVERVIEW	1
1.2 Electrical vehicles	1
1.3 Internet of things (IOT) PLATFORM	3
1.4 Demand prediction	6
1.5 COST OPTIMIZATION	6
1.4 WORK CARRIED OUT	7
CHAPTER 2 LITERATURE REVIEW	8
2.1 Battery monitoring system	8
2.1.1 State of charge (SoC)	8
2.1.2 Determining SoC	9
2.1.3 Coulomb Counting Method	9
2.2 Review of optimization	10
CHAPTER 3 Battery swapper design	12
3.1 Advantages of battery swapping mechanism	12
3.2 Challenges with the battery swapping mechanism	12
3.2 Initial Designs	13
3.4 Finalized design	16
3.5 Future modifications	20
CHAPTER 4 Prototype implementation	21
4.1 Material selection	21
4.2 Engineering drawings	23
4.3 Workshop progress	26
CHAPTER 5 Battery state estimator	29
5.1 Introduction	29
5.2 Kalman filter	29
5.2 Review of Extended kalman filter	29
5.3 Extended Kalman Filter for SoC Estimation	29
5.4 SOC Estimation Based on EKF	30

5.5 State of health (SoH)	31
CHAPTER 6 Mobile application integration	32
6.1 Introduction	32
6.2 Mobile application for users	32
6.3 Mobile application for battery monitoring	32
6.3 Java Code for Extended Kalman Filter (EKF) Algorithm	35
6.4 Conclusion	38
CHAPTER 7 Demand Prediction of a Battery Swapping Station using Time series data statistics	39
7.1 selection of a suitable discrete distribution model	39
. 7.1.2 Log normal distribution	39
7.2 methodology	40
7.3 Data Source	42
7.4 RESULTS	42
7.5 Conclusion	44
CHAPTER 8 Demand forecasting for a battery swapping station using ARIMA modelling	45
8.1 Data gathering	45
8.2 Autoregressive integrated moving average	45
8.3 Methodology	45
8.4 Model checking	49
8.5 Conclusion	51
CHAPTER 9 Optimization	52
9.1 Introduction	54
9.2 Lavenberg Marquardt algorithm	53
9.3 Results	54
9.4 Discussion	54
9.5 Conclusion	57
CHAPTER 10 Replacement of the Buck Boost converter	52
10.1 Introduction	53
10.2 Converter setup	54
10.3 Testing of the two Buck Boost converters	64
10.4 Observations and results	86
10.5 Conclusions	89
CHAPTER 11 Problems encountered	90
CHAPTER 12 Conclusions	91
References	92

LIST OF FIGURES

Figure 1.1	Impacting factors for electrical vehicles	1
Figure 1.2	plugged charging method for electric vehicle	2
Figure 1.3	Wireless chagrining method for electric vehicle	2
Figure 1.4	User case diagram for the battery swapping station	3
Figure 1.5	Grant chart of the project activities	8
Figure 2.1	A BMS framework	9
Figure 2.2:	How gradient decent algorithm reach the minimum value	12
Figure 3.1	Dock type battery tray made using ss rods and connector	14
Figure 3.2	Dock type battery tray made using L-bars and connector	14
Figure 3.3	Casing for the battery	15
Figure 3.4	Casing for the battery front view	15
Figure 3.5	Front view of the design v3 slider dock mechanism	16
Figure 3.6	Rear view of the design v3 slider dock mechanism	16
Figure 3.7	Interior of design v3 slider dock mechanism	17
Figure 3.8	design v3 slider dock mechanism with batteries inserted	17
Figure 3.9	finalized design for the battery swapper	18
Figure 3.10	Outer Case design for the battery swapper	18
Figure 3.11	SLIDING MECHANISM	19
Figure 3.9 :	SLIDING MECHANISM ATTACHED	19
Figure 3.13	Ball bearings	19
Figure 3.14	Inner frame CAD design	20
Figure 3.15	Rear view of the inner frame with Anderson connection	20
Figure 3.16	Swapping mechanism with super capacitor module added	21
Figure 4.1	Removable battery clamps with isolation	22
Figure 4.2	Anderson connectors	23
Figure 4.3	Anderson connected to the casing with the wiring attached	23
Figure 4.4	Engineering drawing for the inner frame	25
Figure 4.5	Engineering drawing for the sliders	25
Figure 4.6	Engineering drawing for the outer case	26
Figure 4.7	Engineering drawing for the battery case	26
Figure 4.8	Engineering workshop progress I	27
Figure 4.9	Engineering workshop progress II	27
Figure 4.10	Engineering workshop progress III	27
Figure 4.11	Engineering workshop progress IV	28
Figure 4.12	Engineering workshop progress V	28
Figure 4.13	Engineering workshop progress VI	28
Figure 4.14	Engineering workshop progress VII	29
Figure 4.15	Engineering workshop progress VIII	29
Figure 4.16	Engineering workshop progress IX	29
Figure 5.1	Diagram of EKF based SOC estimation algorithm	30
Figure 5.2	Block diagram of EKF	31
Figure 5.3	General diagram of the extended Kalman filter	32
Figure 6.1	Integration of the raspberry pi-based environment into android	33
Figure 6.2	Android studio interface	34
Figure 6.3	Bluetooth interface of the android application	35
Figure 6.4	Interface of the SoC and SoH display	35
Figure 6.5	Interface of the swapping station display	36

Figure 7.1.	Variation of the SMA with the time on Monday	43
Figure 8.1.	Time series data for 9 consecutive Mondays without a trend	47
Figure 8.2.	Time series data for 9 consecutive Mondays with a trend	48
Figure 8.3.	Time Series plot with the trending removed	48
Figure 8.4.	ACF Correlograms for the time series data without a trend with 95% confidence intervals	49
Figure 8.5.	ACF Correlograms for the time series data with a trend with 95% confidence intervals	49
Figure 8.6.	Historical data and the forecast for the ARIMA(1,0,1) without trend	50
Figure 8.7.	Historical data and the forecast for the ARIMA(1,0,1) with trend using one iteration	51
Figure 9.1	Sigmoid function	54
Figure 9.2	MATLAB code for the objective function	55
Figure 9.3	Daily variation of price per battery using grid & standby generator (C_s) & sales price	56
Figure 9.4	Total cost per day variation with the number of iterations	56
Figure 9.5	variation of $\alpha(k)$ throughout the day	57
Figure 9.6	Variation of the fully charged batteries throughout the day	57
Figure 9.5	Number of batteries used to transfer energy with the grid	58
Figure 10.1	Input signal (blue) and the output signal (Yellow)	58
Figure 10.2	Circuit design for the Optocoupler circuit	59
Figure 10.3	PCB Design for the optocoupler circuit	59
Figure 10.4	Circuit made using CNC	60
Figure 10.5	Created Negative for the UV method	60
Figure 10.6	Dot Board Circuit	61
Figure 10.7	SOLIDTHINKING EMBED code for the PWM modification	61
Figure 10.8	Outputs of the two optocouplers	62
Figure 10.9	Observation on the control board I	62
Figure 10.10	Observation on the control board II	63
Figure 10.11	Observations on PWM inputs of the controller board (Vin1 & Vin2)	63
Figure 10.12	Testing setup of the converters	64
Figure 10.13	Test setup of the converters	65
Figure 10.14	Water load tested	65
Figure 10.15	Launchpad input with 0.5 duty cycle	66
Figure 10.16	Oscilloscope reading	66
Figure 10.17	Current draw reading	67
Figure 10.18	Oscilloscope reading	67
Figure 10.19	current draw reading	68
Figure 10.20	oscilloscope reading	69
Figure 10.21	current draw reading	69
Figure 10.22	oscilloscope reading	70
Figure 10.23	current draw reading	70
Figure 10.24	oscilloscope reading	71
Figure 10.25	current draw reading	71
Figure 10.26	oscilloscope reading	72
Figure 10.27	current draw reading	72
Figure 10.28	Launchpad input with 0.5 duty cycle	73
Figure 10.29	oscilloscope reading	73
Figure 10.30	current draw reading	74
Figure 10.31	oscilloscope reading	74
Figure 10.32	current draw reading	75
Figure 10.33	oscilloscope reading	75
Figure 10.34	current draw reading	76
Figure 10.35	oscilloscope reading	76
Figure 10.36	current draw reading	77
Figure 10.37	oscilloscope reading	77

Figure 10.38	current draw reading	78
Figure 10.39	oscilloscope reading	78
Figure 10.40	current draw reading	79
Figure 10.41	Oscilloscope waveform for the buck mode without capacitor	79
Figure 10.42	Current rating buck mode without capacitor	80
Figure 10.43	Oscilloscope waveform for the buck mode	80
Figure 10.44	Current rating buck mode	81
Figure 10.45	Oscilloscope waveform for the buck mode	81
Figure 10.46	Current rating buck mode	82
Figure 10.47	Oscilloscope waveform for the buck mode	82
Figure 10.48	Current rating buck mode	83
Figure 10.49	Oscilloscope waveform for the buck mode	83
Figure 10.50	Current rating buck mode	84
Figure 10.51	Oscilloscope waveform for the buck mode	84
Figure 10.52	Current rating buck mode	85
Figure 10.53	Oscilloscope waveform for the buck mode	85
Figure 10.54	Current rating buck mode	86
Figure 10.55	Variation of Input current with the Water Load resistance	87
Figure 10.56	Variation of Output Voltage with the Water Load resistance	87
Figure 10.57	Variation of Output current with the Water Load resistance	88
Figure 10.58	Variation of Temperature with the Water Load resistance	88

LIST OF TABLES

TABLE 4.1	Gauge chart for the copper wires	22
TABLE 4.2	Bill of Quantities for the prototype	24
TABLE 7.1.	Generated time series data for the Monday	43
TABLE 7.2	SMA estimate and the MSE for the generated dataset	44
TABLE 7.3	parameter estimation for the different distributions	44
TABLE 7.4	Pdf for the different distributions for 2-3pm interval	44
TABLE 7.5	Error indices for the distribution models	45
TABLE 7.6	Confidence intervals for the distribution models	45
TABLE 8.1.	Time series data for 9 consecutive Mondays with a trend	47
TABLE 8.2	Estimated parameters for different models for the time series data without a trend	51
TABLE 8.3	Estimated parameters for different models for the time series data with a trend	52
TABLE 8.4	goodness of fit tests	52
TABLE 8.5	Forecasted demand for the next time steps	52
TABLE 10.1	Specifications of the Buck Boost converter	58
TABLE 10.2	Observations of the Load Testing	86

LIST OF ABBREVIATIONS

ACF	Autocorrelation function
AIC	Akaike information criteria
API	Application programming interface
BEV	Battery Electric Vehicle
BIC	Bayesian information criterion
BMS	Battery Management system
BOQ	Bill Of Quantities
BSS	BATTERY SWAPPING STATION
DOD	Depth of Discharge
EKF	Extended Kalman Filter
EV	Electrical vehicles
GD	Gradient decent
GI	Galvanized Iron
HV	Hybrid Vehicle
IDE	integrated development environment
IoT	Internet of Things
LMA	Levenberg–Marquardt algorithm
MAPE	Mean Absolute Percentage error
MSE	Mean Square Error
PACF	Partial Autocorrelation Function
PHEV	Plug-in Hybrid Electric Vehicle
PV	Photo voltaic
SG	Standby generator
SMA	Simple Moving Average
SOC	State of Charge
SOH	State of Health
QoS	Quality of Service

CHAPTER 1 INTRODUCTION

1.1 OVERVIEW

This project was targeted to build a complete IoT based business eco system where this eco system can be used for the purpose of management, supervision and control of various stakeholders of this eco system. For this purpose, we have tasked with the following objectives.

- Design of a practically feasible hardware prototype for an electrical vehicle battery swapping mechanism
- Implementation of this hardware prototype
- Design and upgrade the IoT platform associated with the electrical vehicles and the battery swapping station.
- Integration of the Extended Kalman filter based battery state estimating algorithm into a mobile platform
- Develop a demand prediction model for the battery swapping station
- Develop a profit maximization algorithm to utilize the energy sources in the battery swapping station.

1.2 ELECTRICAL VEHICLES

Electrical vehicles are the vehicles that runs on electrical powered motor instead of diesel or petrol based mechanical engine. Recently due to technological developments, EV's becoming more and more popular in the market. It is expected that by 2030 the total electric vehicles used globally would reach up to 200million.

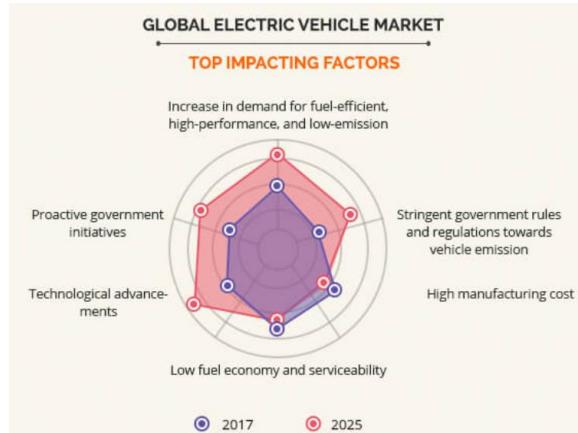


Figure 1.1 Impacting factors for electrical vehicles*

Due to the environmental impact of the petroleum-based vehicles and due to fact that petroleum is nonrenewable energy source, research based on electric based vehicles is becoming more and more popular. At present there is a huge demand for electric cars such as Telsa model X ,BMW i3 ,Chevrolet Volt EV has much demand in USA and other European countries.

* Image credit: <https://www.alliedmarketresearch.com/electric-vehicle-market>

1.2.1 Electrical vehicle charging methods

Typical electric vehicle consists of a battery pack, and it needs to be recharged once the battery SOC is depleted. There are multiple EV charging methods available, some are commercially available and some are under research. Some available charging methods are,

- Wireless charging
- Via Charging cable that connected to local grid
- Battery swapping

Most commonly used electrical chagrining technique uses charging ports that acts as the chagrining infrastructure, but the most common issue with this method is the time it takes to charge a vehicle. Typically, it takes more than 30 minutes. And if there is a queue in the charging port, then the customers have to wait much longer.



Figure 1.2 plugged charging method for electric vehicle

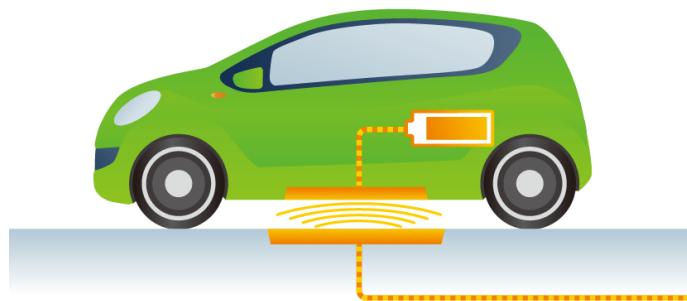


Figure 1.3 Wireless chagrining method for electric vehicle

1.2.2 Battery swapping mechanism

Battery swapping mechanism is simply a mechanism that can replace the drained battery with a fully charged one, there are swapping mechanisms such as

- Automatic battery swapping mechanisms
- Semi manual Battery swapping mechanisms
- Manual Battery swapping mechanisms

1.2.3 Battery Swapping Station (BSS)

Battery swapping station is a service station where the EV's can replace their charged depleted batteries into fully charged batteries. The battery swapping station is capable of charging batteries via multiple energy sources. The BSS can also supply energy back into the grid to maximize the income received through managing the batteries.

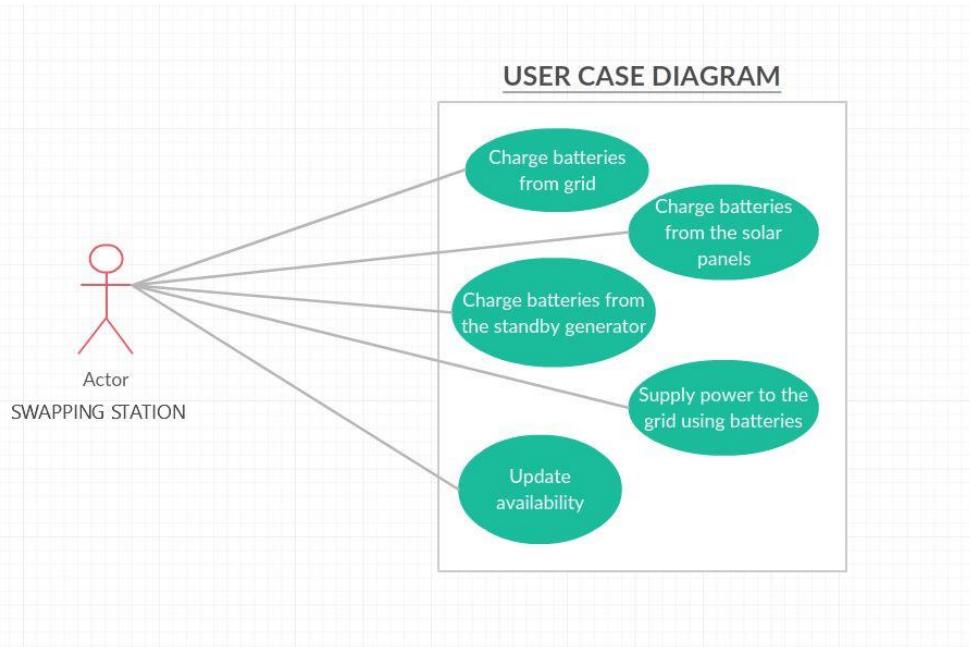


Figure 1.4 User case diagram for the battery swapping station

The battery swapping station is integrated into an IoT platform, so all the customers can monitor the availability of batteries and reserve batteries before arriving at the swapping station.

1.3 INTERNET OF THINGS (IOT) PLATFORM

Remote IoT platforms are becoming more and more popular in the engineering industry as they gather data from remote sources, make meaningful decisions based on the gathered data such as management provisioning and automation of services associated with the connected devices.

1.3.1 Applications of IoT platforms

IoT platforms can be used with smart devices to perform remote control tasks, remote monitoring functions and integrate with already existing systems to increase their efficiency. IoT platform can also be used to optimize cost of product and service-based businesses. For example, a business can gather data from sensors to perform production analysis to optimize the cost of production.

Smart city concept is also becoming popular in the present which targets to build IoT based infrastructure to improve efficiency of the government services such as healthcare, travelling, traffic control etc.

1.3.2 Commonly available IoT platforms

➤ Google Cloud IoT

Google Cloud Platform is a collection of public cloud services offered by google. Services provided by google IoT platform include computation services, storage services and provide gateway for services provided by the google. With the google IoT platform, a developer can build and manage IoT systems. Google IoT consists of the following main components.

- **Cloud IoT Core** - Allows connecting various devices and gathers their data.
- **Cloud Pub/Sub**- Processes event data and provides real-time stream analytics.
- **Cloud Machine Learning Engine** - allow building of Machine Learning models

The IoT solution developed by Google includes a number of other services that may be useful while building comprehensive connected networks.

➤ Oracle Internet of Things

In comparison of the top Internet-of-Things platforms Oracle is a major platform. Oracle is well known for their advanced cloud computing, database management and enterprise management software and their related IoT solutions. Oracle platform is known for building commercial apps with flexible environment. Oracle is a viable candidate for a large scale IoT network, and is rich of advanced security mechanisms

➤ IBM Watson Internet of Things

IBM Watson has a vision on the IoT , “the Internet of Things becomes the Internet that thinks” meaning that IBM experiments with integration IoT with artificial intelligence creating unique experiences and solutions.

The IBM Watson IoT platform supports services such as,

- Effective remote device control,
- Secure data transmission and storage in cloud
- Real-time data exchange
- Machine learning options with the integration with AI technology.

The development platform offered by IBM includes a number of convenient tools and services, making IoT software creation easier and more efficient.

1.3.3 Raspberry pi platform

Raspberry Pi(RP) is an minicomputer capable of acting as a computer with all the peripherals attached, Raspberry pi computers capable of running operating system such as Raspbian. RP is popular among do it yourself (DIY) enthusiasts and developers due to the lower price range and portability of the RP. RP can perform calculations and gather data via GPIO (general purpose input/output) pins or via the built-in wireless connectivity.

1.3.4 Android platform

The Android platform is a platform for mobile devices that uses a modified Linux kernel. The Android Platform was introduced in November of 2007 but now being the most popular mobile operating system in the world. Android applications can be written with the help of java programming language.

Android is an open development platform meaning once the source code is released to the public after it is finalized the community can modify the code as they see fit. Any user can develop android applications using the software development kit provided by the Google.

1.3.5 Communication models in IoT

Various communication methods are available that is related to the IoT platforms.

- **The device-to-device communication model**- Two or more devices directly connect and communicate between one another, without using an intermediary application server. This model uses communication protocols such as Bluetooth, Z-Wave, or ZigBee to establish communications.
- **Device-to-cloud communication model** - IoT device connects directly to an Internet cloud service (eg - application service provider) to exchange data and control message traffic. Advantage of this system is it can use existing communication infrastructure(eg- Wifi, Ethernet). The data protocol can be can be proprietary such that the information shared through the internet is secure.
- **Device-to-gateway model** - The IoT device connects through an Application layer gateway service to reach a cloud service. The intermediate software will provide the security and other functionalities required by the IoT device to operate correctly. (example- Home automation systems consist of a hub that acts as the gateway to the cloud services)
- **Back end data sharing model** - This enables users to export and analyze smart object data from a cloud service in combination with data from other sources. So the data gathered from IoT devices can be used by multiple parties to perform unique tasks.

1.3.6 Quality of Service

Quality of Service (QoS) manages the network capabilities and resources to provide a reliable path for the IoT connectivity. To offer secure and predictable services, QoS will manage delays, delay variation, bandwidth and packet loss by classifying traffic and registering channel limits. With effective QoS management, the IoT system will have a much better chance of receiving warnings or high priority messages in real-time.

1.3.7 GPS based location tracking of EV

Mobile phone tracking is a process for identifying the location of a mobile phone, whether stationary or moving with the help of global positioning system. Same concept can be used to identify the location of the electric vehicle, which is required by the BSS to predict the future demand. The tracking requires a gps antenna and a receiver on the EV side and a communication method such as internet or network connectivity that allows the EV to send its current coordinates to the server

1.4 DEMAND PREDICTION

Demand prediction is the mathematical analysis that tries to understand the behavior of the customers and predict the future demand based on that behavior

For better management of the available batteries in the swapping station, it is required to predict the upcoming demand at least few hours beforehand. To predict the demand, two methods were proposed in this project.

- By using statistical model fitting
- By using ARIMA modelling

From the point of view from the swapping station, the BSS needs to maintain a stock of fully charged batteries to meet the demand of the vehicles that are arriving at the BSS. Furthermore, the BSS can sell energy back to the grid to make more profit from the batteries, when the future demand for the batteries are low. For this purpose, statistical based prediction of the future demand is proposed taking into the consideration of time series data.

The demand for the batteries from electrical vehicle customers depends on several factors

- State of charge of the Customer
- Time of the day
- Geographical location of the BSS
- Geographical location of the electrical vehicle

Attention needs to be given for the demand variation with time, because the demand can consist of a trend that varies with the vehicular traffic around the BSS. When considering typical vehicular traffic in urban area, during morning hours before and after, the schools and office starts, and during the lunch breaks the demand would be expected to be higher. And for the off-peak hours during early morning hours, the demand is expected to be lower.

There are various methods involved in analyzing time series data to predict the future trend available, moving average models such as ARMA, ARIMA based forecasting, Machine learning based methods etc.

1.5 COST OPTIMIZATION

To earn maximum profit from the BSS, optimization of the energy sources and charging decisions for the batteries need to be implemented. By day ahead planning for the BSS, the resources of the BSS can be utilized in an efficient manner.

1.4 WORK CARRIED OUT

In order to meet the objectives, the following activities were carried out according to the schedule represented by the Gantt chart in Figure 1.5.

- Activity 1 – Literature survey
- Activity 2 – Development of IoT device
- Activity 3 – Testing and verification of IoT device
- Activity 4 – Development of battery swapper mechanism
- Activity 5 – Developing Algorithms for the battery swapper

Activities	Week 1-2	Week 3-4	Week 5-6	Week 7-8	Week 9-10	Week 11-12	Week 13-14
Literature Survey	X						
Development of IoT Device				X	X		
Testing and Verification of IoT device					X	X	X
Development of Battery swapper mechanism		X	X				
Developing Algorithms for battery swapper					X	X	X

Figure 1.5 Gantt chart of the project activities

CHAPTER 2 LITERATURE REVIEW

2.1 BATTERY MONITORING SYSTEM

Battery Monitoring System (BMS) is a smart system whose function is to monitor the vigor of a battery pack. BMS computes the battery's capacity, depreciation of battery while the charging/discharging and correct productivity of the battery and provides this information in real time to users. This mitigates the sense of incorrect safety of periodic battery assessment as it is vigilant to emerging issues before hand the occurrence of a possible malfunction. As every cell is observed separately, so any damage can be checked and appropriate warnings against the values pre-set by consumers and protective measures can be employed, safeguarding the other cells against cumulative damage thereby extending battery life. BMS logs history data of all measured parameters for further analysis and future reference.

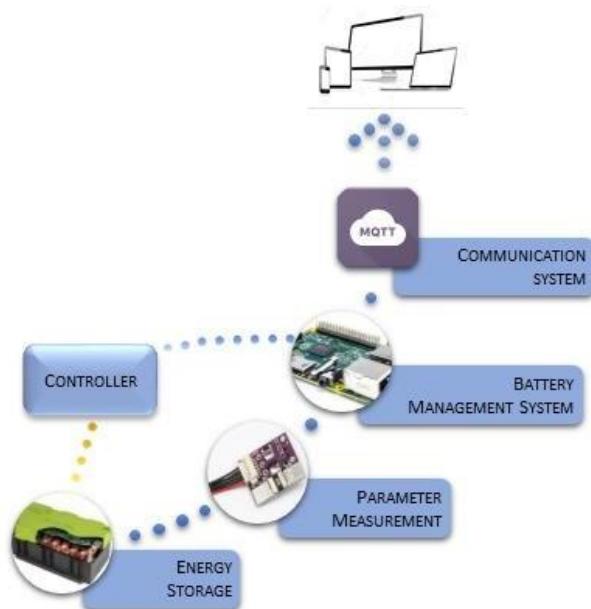


Figure 2.1 A BMS framework

Figure 2.1 shows a typical BMS framework incorporating the measurement of key battery parameters e.g., current, voltage, temperature etc., and performing necessary calculations/estimations to extract useful information about energy storage system i.e., State of Health (SoH), State of Charge (SoC), operating temperature range. Based on these calculated parameters, controlling actions are taken to maintain the battery's lifecycle and safety against potential hazards. Therefore, the prime objective of monitoring is to gauge various variables, log events, generate warnings, record usage profile and represent this information locally and remotely to the user.

BMS is unable to sense movable connections present in the battery, leakage of cell material, corrosion of connections leading to the development of high resistance and subsequently fire danger. It is also unable to visually monitor developing swelling, potential leakage, cracks in the outer geometry of battery pack etc.

2.1.1 STATE OF CHARGE (SOC)

State of charge (SoC) is a critical factor to guarantee that a battery system is operating in a safe and reliable manner. Many uncertainties and noises, such as fluctuating current, sensor measurement accuracy and bias, temperature effects, calibration errors or even sensor failure, etc. pose a challenge to the accurate estimation of SOC in real applications.

State of charge is the equivalent of a fuel gauge for the battery pack in a Battery Electric Vehicle (BEV), Hybrid Vehicle (HV), or Plug-in Hybrid Electric Vehicle (PHEV). The units of SoC are percentage points (0% = empty; 100% = full). An alternate form of the same measure is the **Depth of Discharge (DoD)**, the inverse of SoC (100% = empty; 0% = full). SoC is normally used when discussing the current state of a battery in use, while DoD is most often seen when discussing the lifetime of the battery after repeated use.

2.1.2 DETERMINING SOC

Usually, SoC cannot be measured directly but it can be estimated from direct measurement variables in two ways: offline and online. In offline techniques, the battery desires to be charged and discharged in constant rate such as Coulomb-counting. This method gives precise estimation of battery SoC, but they are protracted, costly, and interrupt main battery performance. Therefore, researchers are looking for some online techniques. In general there are five methods to determine SoC indirectly:

- Chemical
- Voltage
- Current integration
- **Kalman filtering**
- Pressure

To overcome the shortcomings of the voltage method and the current integration method, a Kalman Filter can be used. The battery can be modelled with an electrical model, which the Kalman filter will use to predict the over-voltage, due to the current. In combination with coulomb counting, an accurate estimation of the state of charge can be obtained. The strength of a Kalman filter is that it is able to adjust its trust of the battery voltage and coulomb counting in real time. The SoC can be defined as follows,

$$SoC(t) = \frac{Q(t)}{Qn}$$

Where,

$$\begin{aligned} &\text{Current capacity - } Q(t) \\ &\text{Nominal capacity - } Qn \end{aligned}$$

The battery manufacturer specifies the nominal capacity which shows the utmost quantity of charge that can be stored in the battery.

2.1.3 COULOMB COUNTING METHOD

With assumption that initial SoC (at time t_0) is in knowledge, SoC at any instant is usually estimated by integration of the battery current over time, as shown in equation,

$$SoC(t) = SoC(t_0) + \int_{t_0}^{t_0+\Delta t} \frac{I_{bat} d\Delta t}{Qn} \times 100\%$$

Where,

$$\begin{aligned} I_{bat} &: \text{value of battery current,} \\ Q_n &: \text{nominal capacity} \end{aligned}$$

The accurateness of Coulomb counting technique depends upon various parameters viz., operating temperature, battery usage history, discharge current, and cycle life. The coulomb counting technique consists of using the equation by enumerating the charge supplied by the battery by sensing its input and output current. Though, few inefficiencies are there in this technique - the initial SoC value is not correctly known, presence of self-discharge phenomena can change the real SoC value after a prolong storage time and battery degradation due to aging should be taken into consideration.

2.2 REVIEW OF OPTIMIZATION

2.2.1 Algorithm

Algorithm is a set of rules that is followed by step by step method of solving problem, it is commonly used in data processing, calculations and other fields. Usually algorithms manipulate data in various ways to perform a certain task.

2.2.2 Iterative method

In computational mathematics, an **iterative method** is a mathematical procedure that uses an initial guess to generate a sequence of improving approximate solutions for a class of problems, in which the n^{th} approximation is derived from the previous $n-1$ terms.

2.2.3 Numeric optimization algorithms

Numerical optimization involves generating a sequence of estimates for the solution, that will arrive at a solution (or sufficiently close to a solution) methods are usually available to find the best solution of a problem. Examples for numerical (optimization) minimization algorithms include,

- Gradient descent algorithm
- Steepest descent algorithm
- Newton method
- Lavenberg Marquardt algorithm

2.2.4 Objective function

Objective function is the mathematical equation that describe the intended output of the target. And this equation can be altered to minimize or maximize the cost function by changing its variables.

2.2.5 Numeric optimization algorithms

Usually there are two types of numerical optimization methods.

- If the objective function is concave (maximization problem)
- If the objective function is convex (minimization problem)

2.2.6 Constrained Optimization

Constrained optimization is the achieving the objective function with respect to some variables, but the variables may possess a constraint on them. For example, an objective function can have constraints such as capacity, availability, cost limitations etc. These constraints can be,

- Hard constraints - Variables needs to be satisfied these constraints
- Soft constraints - The variables constrained are not satisfied then the objective function is penalized

2.2.7 Global minima vs local minima

The minimum value obtained for a given range is called a local minima because there is a possibility of other minimum values outside of these specified range. The minima value is global minima if it is the lowest possible level attainable in the entire domain of the given objective function.

2.2.8 Gradient decent algorithm

Gradient decent algorithm is a first order iterative optimization algorithm which is used to find the minima of a function. Stochastic gradient decent can be used to minimize the number of iterations plus computing power; when new data is introduced. This method uses only one set of data or a mini batch when finding the grad operator on the depended parameters.

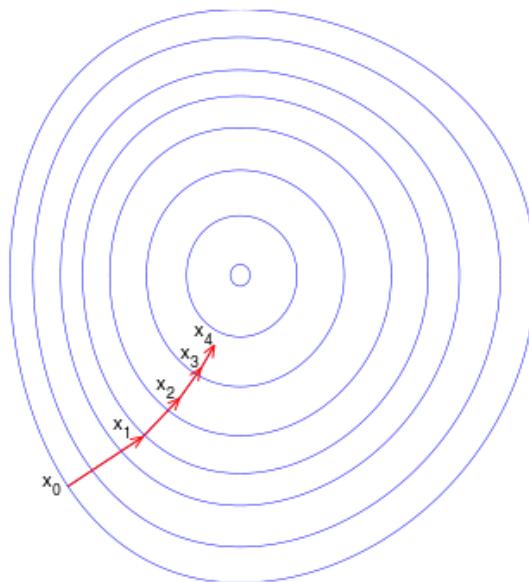


Figure 2.2: How gradient decent algorithm reach the minimum value

In gradient decent the sum of squared errors is reduced by updating the parameters in the steepest descent direction.

If a multi variable function $F(x)$ is defined in a neighbourhood of point a_n . then $F(x)$ decreases fastest if the function goes from a_n in the direction of the negative gradient of F ,

$$a_{n+1} = a_n - \gamma \nabla F(a_n)$$

The resulting would be $a_{n+1} \leq a_n$ for all n positive values. (monotonically decreasing)

γ is the step rate (Learning rate) which is the allowance of change allowed for each iteration.

2.2.9 Lavenberg Marquardt algorithm

Levenberg-Marquardt is a commonly used iterative algorithm to solve non-linear minimization problems.it can be considered as a combination of gradient decent and gauss newton method, where if the algorithm acts as gradient decent algorithm when the parameters are far to the optimal value. And act like gauss newton method when the parameters are close to the optimal value.

The algorithm will reach the local minima with number of iterations depending on the initial guess of the parameter values

CHAPTER 3 BATTERY SWAPPER DESIGN

In this part of the project a prototype draft is designed for the battery swapping mechanism which needs to have the following features

- A mechanism to swap drained / batteries that needs recharging
- Practically implementable mechanism

3.1 ADVANTAGES OF BATTERY SWAPPING MECHANISM

- **Takes less time**

With a battery swapping mechanism the time it takes to recharge a battery, will be much less than the alternative methods that are currently being used. EVs normally take 8 hours to charge and up to 1 to 2 hours for a fast charge.

- **Convenient with an IOT system**

With an IOT system implemented to collect battery data from the consumers, a business model can be strategized to make more efficient swapping station.

3.2 CHALLENGES WITH THE BATTERY SWAPPING MECHANISM

- **Battery standards**

Every manufacturer uses different battery form factors, standards and technologies making a universal swapping mechanism more difficult unless an industry standard is implemented in the future.

- **Cost of implementing battery swapping stations**

The implementation of BSS covering many physical locations can be challenging, and for a location with a more swapping needed (or rush hours), charging of the drained batteries can be challenging. This can be overcome by using forecasting methods, and battery swapping within swapping stations (via transporting charged batteries). Still, dramatic number of vehicles can be recharged within a given time by using swapping method when compared to the port charging method and other alternatives.

- **Performance guarantee of the batteries**

The batteries even with the same brand may have different performance which may result in reduced charge retention and other issues, so replacing these kinds of batteries with good quality batteries can be unfavorable for the users and the business, With the proposed IoT system, the SoH, SOC can be determined to mitigate this issue.

3.2 INITIAL DESIGNS

The battery swapping mechanisms designed such a way that they can be replaced manually or by using semi manual mechanism.

3.3.1 Design v1

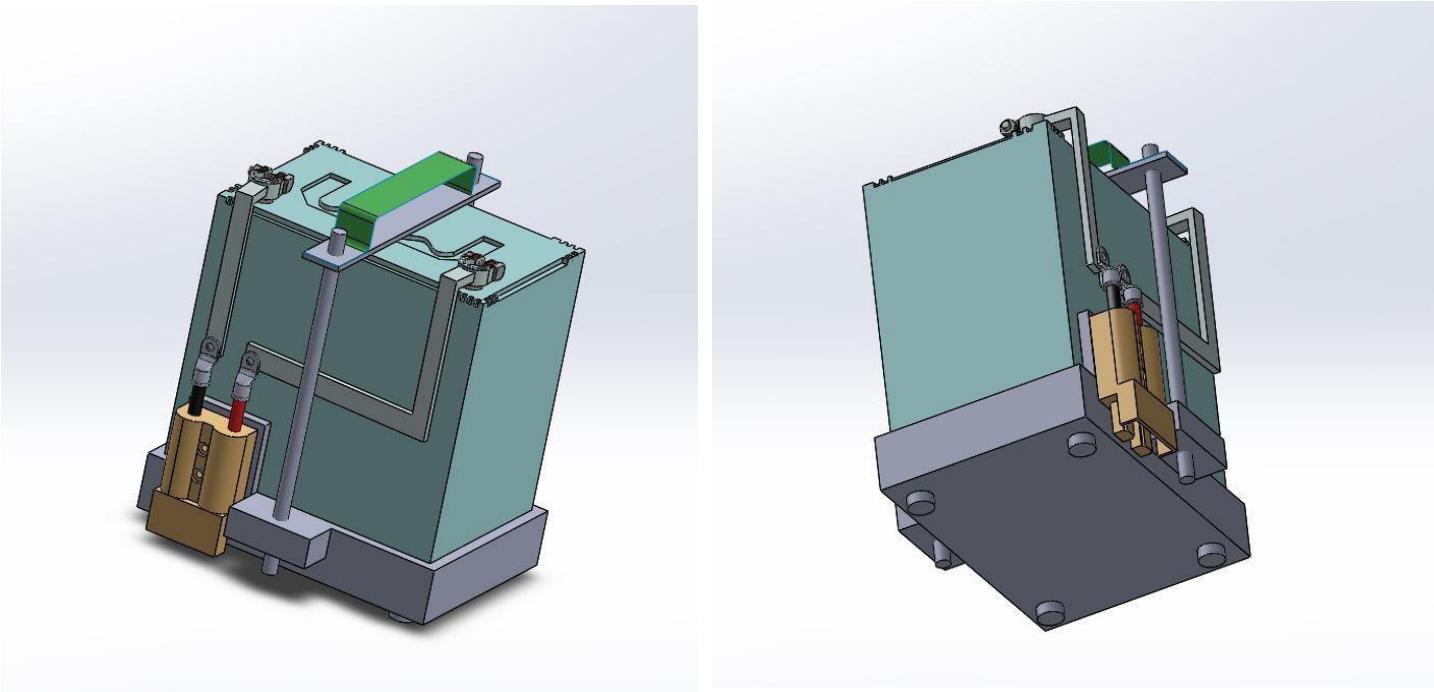


Figure 3.1 Dock type battery tray made using ss rods and connector

3.3.2 Design v2

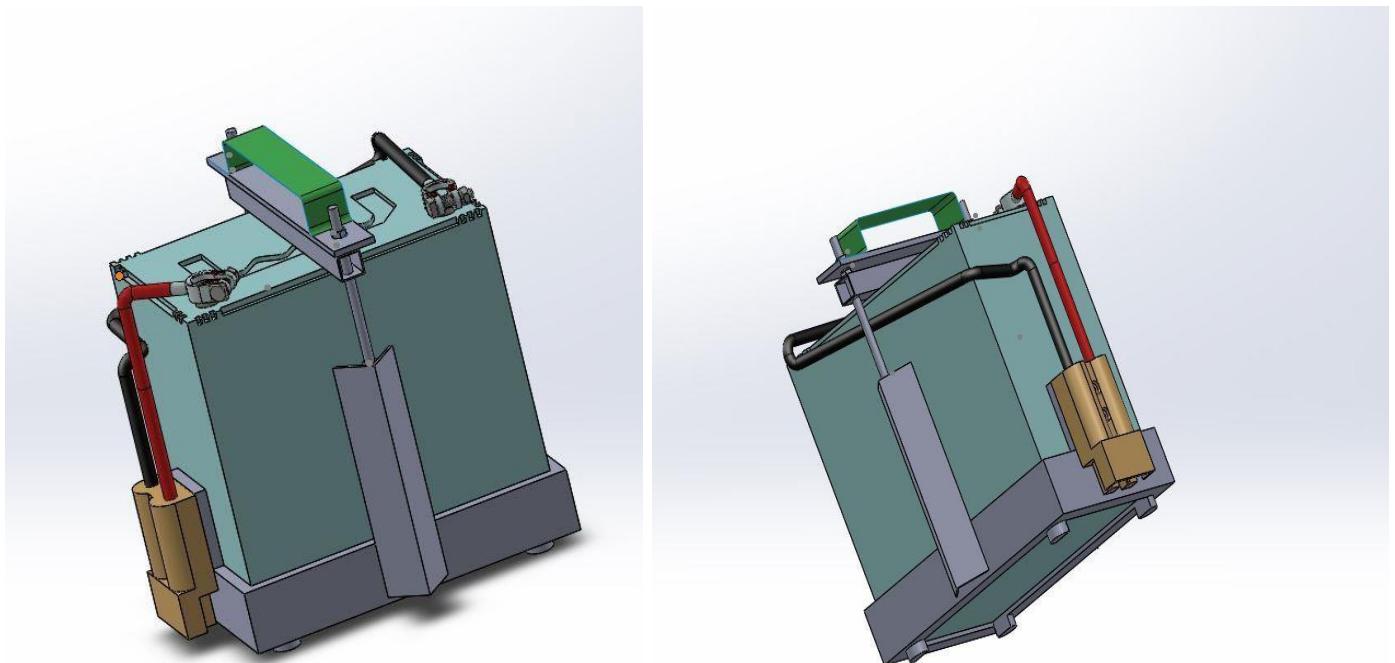


Figure 3.2 Dock type battery tray made using L-bars and connector

Design v2 uses commonly available materials in the local market. Also the implementation and cost of materials is much lower compared to the other design approaches that were considered.

3.3.3 Design v3 – Slot type battery swapping mechanism

This method uses a battery case with a connector which can be docked into the vehicle and the charging unit with the plug (which can stand up to 350A ,600V)

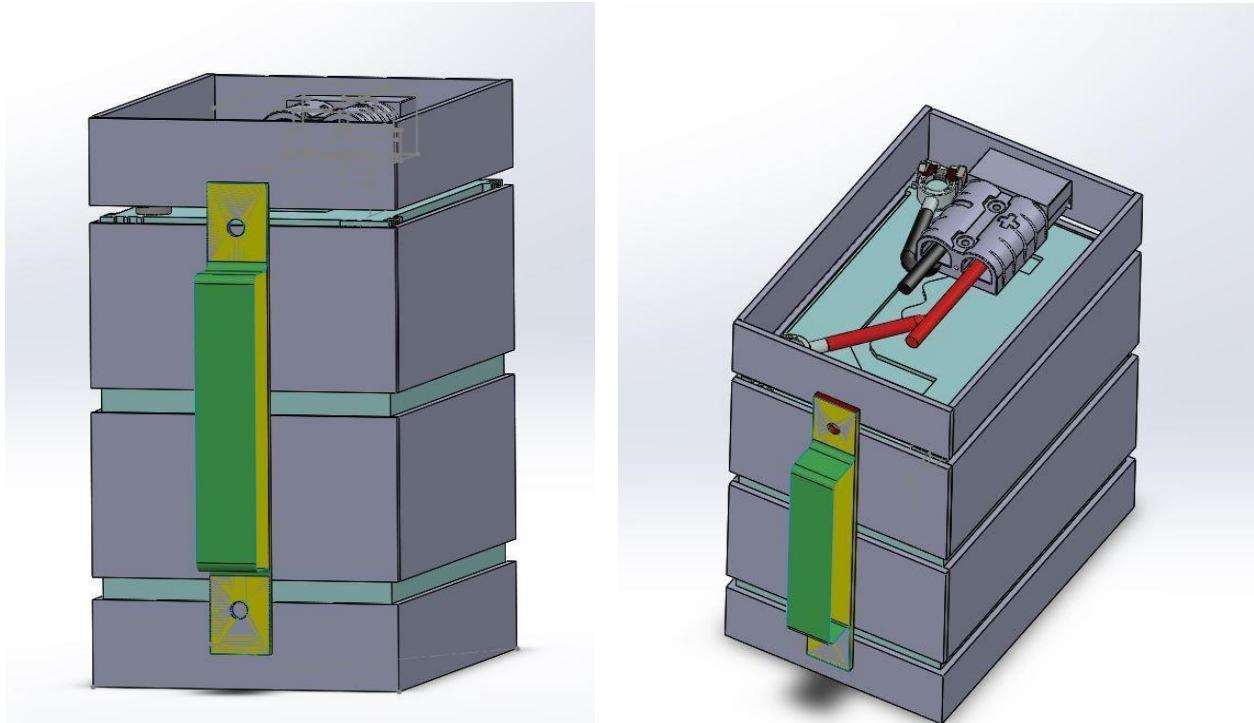


Figure 3.3 Casing for the battery

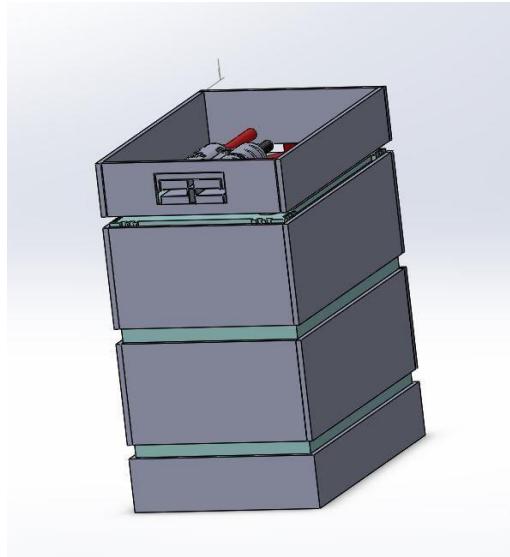


Figure 3.4 Casing for the battery front view

The batteries can be docked into a sliding mechanism, and the batteries are connected in series as shown in the figure 3.6. The sliding mechanism consist of sliders made by using free axis rotating bearings mounted on a GI tubing that allows smooth operation of the sliding mechanism, and for the electrical connection of the batteries is done using Anderson connectors, which can handle up to 175A of current through its point of contact.

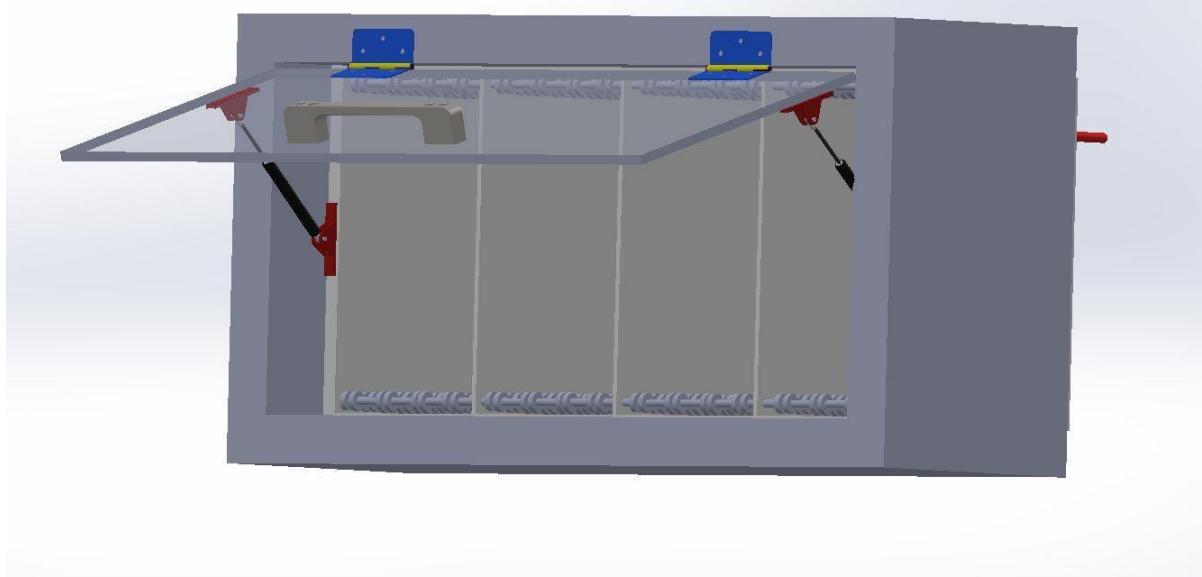


Figure 3.5 Front view of the design v3 slider dock mechanism

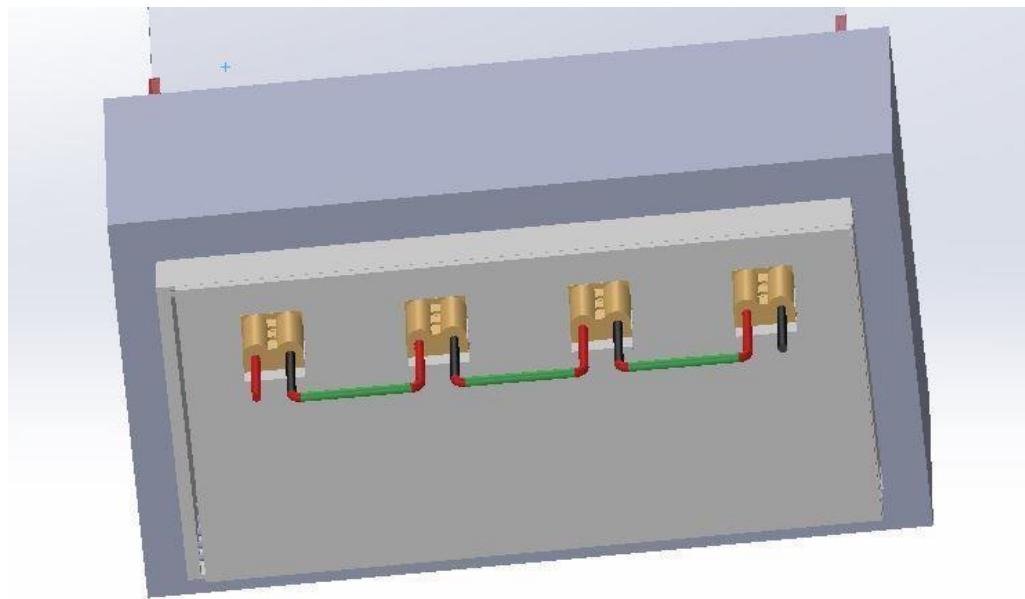


Figure 3.6 Rear view of the design v3 slider dock mechanism

This mechanism can be used on both the electric vehicle and as the charging dock in the battery swapping station. As shown in the figure 3.5 the slider rollers are implemented on both top and bottom part of the dock ensure battery slides smoothly into the connector and it also ensures that battery will remain tightly fitted while the electrical vehicle is being driven.

The battery dock has a gas spring connected door for easier opening and a safety interlock to ensure the battery is properly fixed into the Anderson connector.

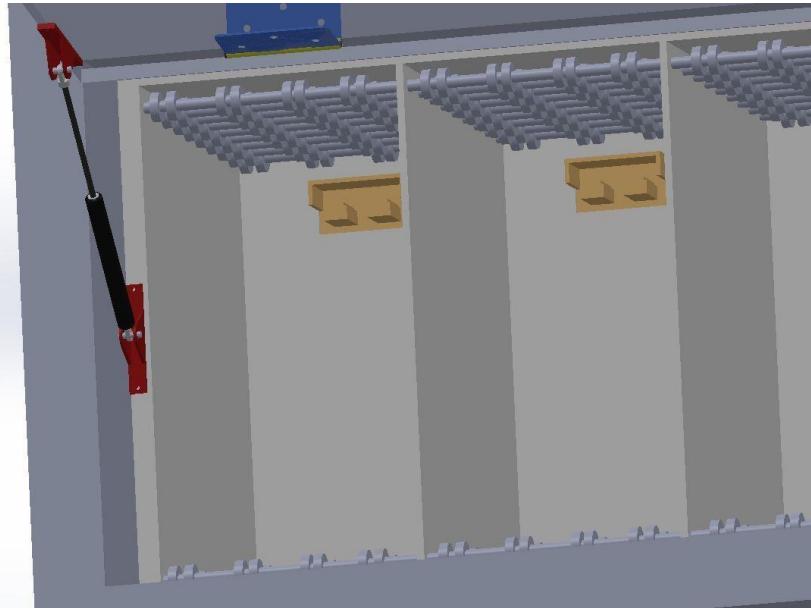


Figure 3.7 Interior of design v3 slider dock mechanism

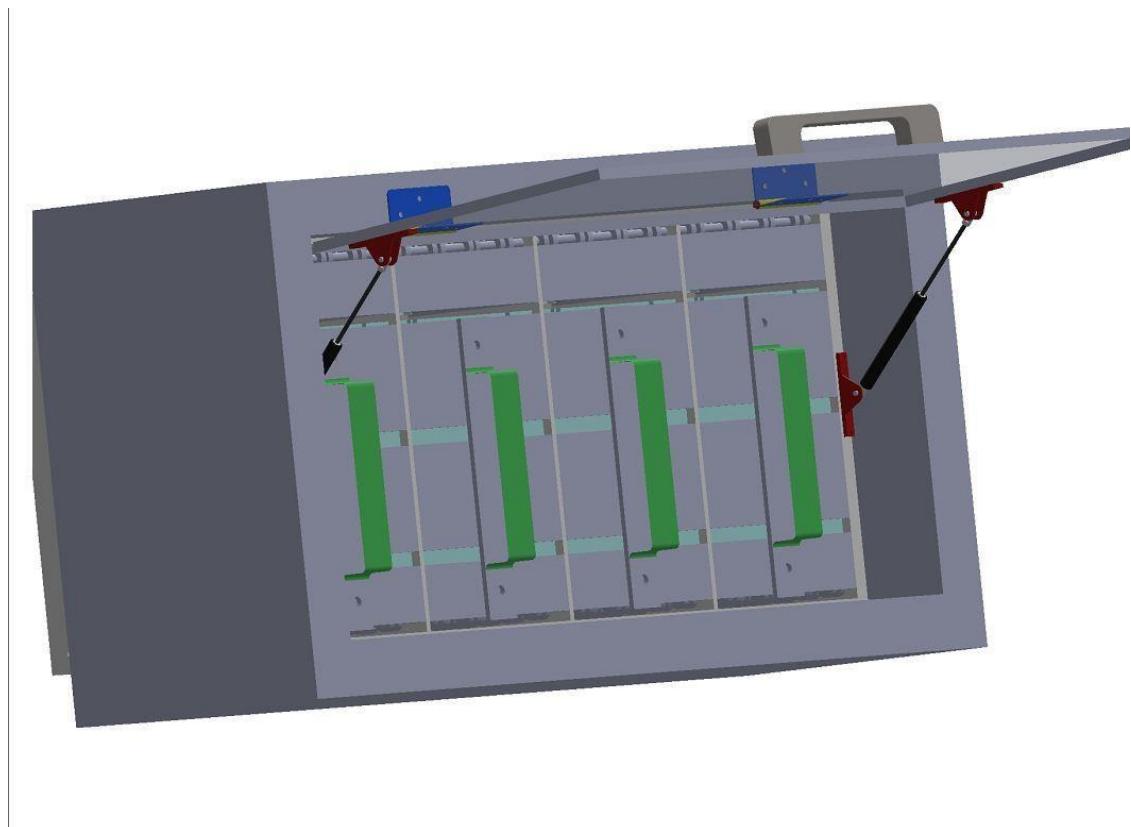


Figure 3.8 design v3 slider dock mechanism with batteries inserted

3.4 FINALIZED DESIGN

After the modifications to the design v3, the finalized design that is to be built is designed, this mechanism consists of

- Battery storing chamber in the vehicle
- Battery moving slider tray
- Battery charging dock (almost same as the storing chamber)

The finalized design for the battery swapping mechanism is shown figure 3.9.

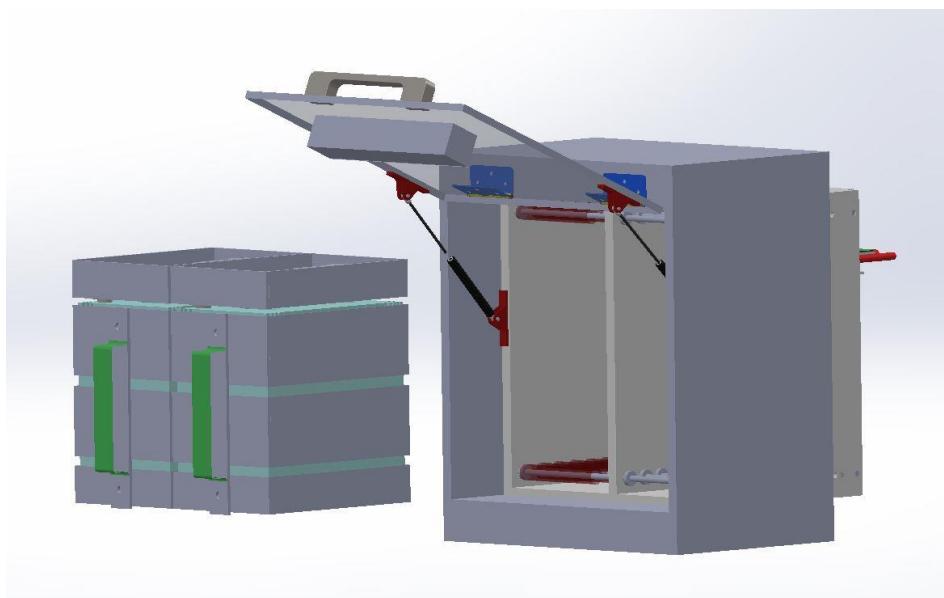


Figure 3.9 finalized design for the battery swapper

3.4.1 Outer frame

The outer frame proposal for the battery swapper is shown in figure 3.10, the proposed mechanism includes a Perspex (transparent) door with gas springs attached for the easier opening, and an interlock mechanism.

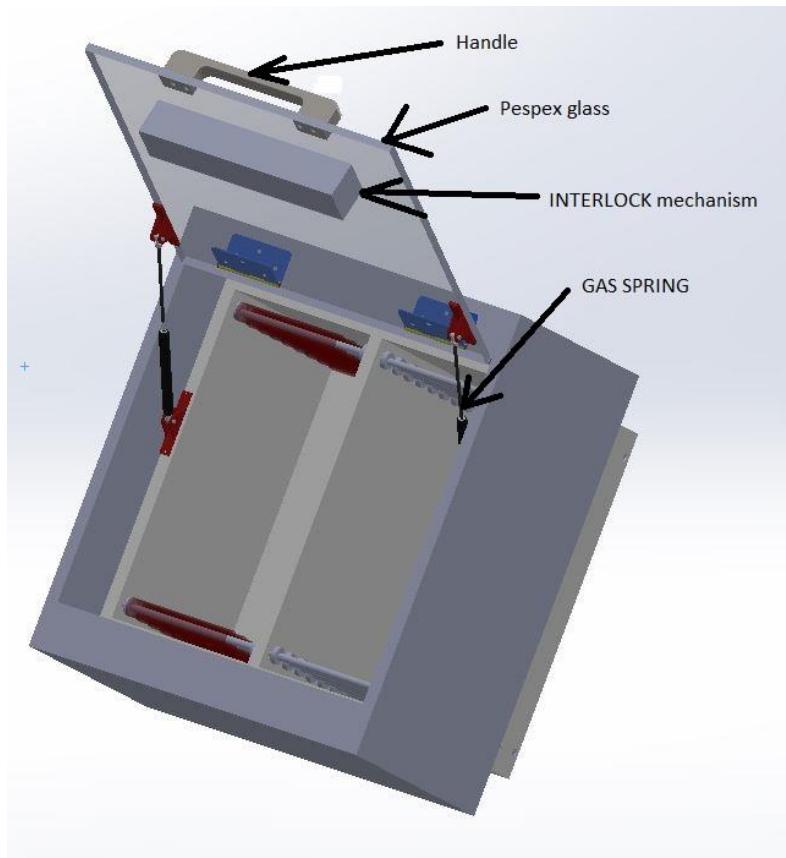


Figure 3.10 Outer Case design for the battery swapper

3.4.2 Sliding mechanism

The sliding mechanism for the battery swapper uses a stainless-steel rod where two ball bearings attached, a GI PIPE is embedded into the bearing which freely rotates around the axis of the stainless-steel rod as shown in figure 3.11. the overall attachment of the sliding mechanism to the internal chamber is shown in figure 3.12 below. Figure 3.13 shows the ball bearing used for the attachment

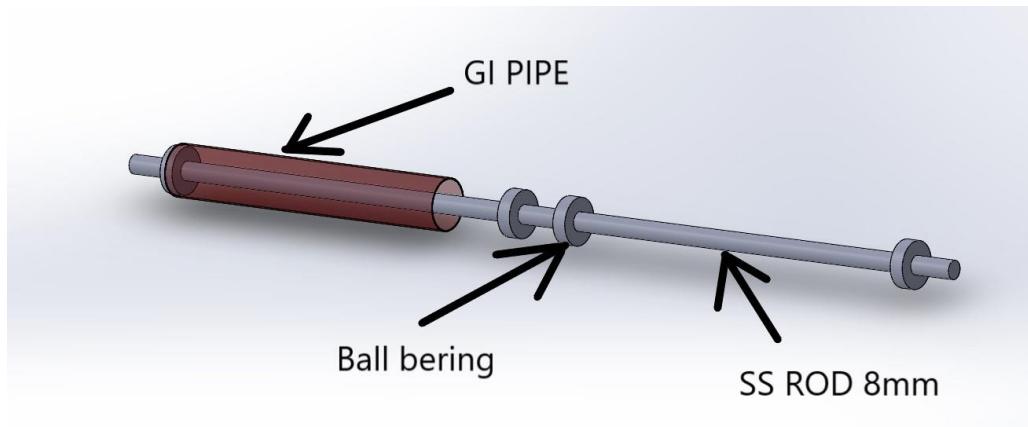


Figure 3.11 SLIDING MECHANISM

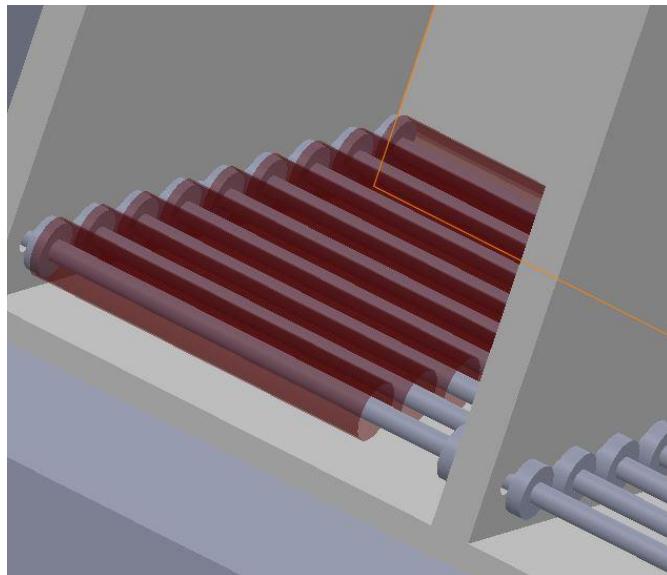


Figure 3.12 SLIDING MECHANISM ATTACHED



Specification: Deep groove sealed miniature ball bearing
Type: 608-2RS
Color: As the picture shows
Size: Approx. 8x22x7mm
Material: Bearing steel
Inner Diameter: 8mm
Outer Diameter: 22mm
Thickness: 7mm

Figure 3.13 Ball bearings

3.4.3 Inner Frame

The inner frame is made with heavy duty box bar ($\frac{3}{4}'' \times \frac{3}{4}''$) to withstand the loading of the batteries and maintain the rigidity of the frame. The inner frame of the mechanism is shown in figure 3.14 ,3.15 below. The Anderson connector is also attached to the inner frame to allow easier connection.

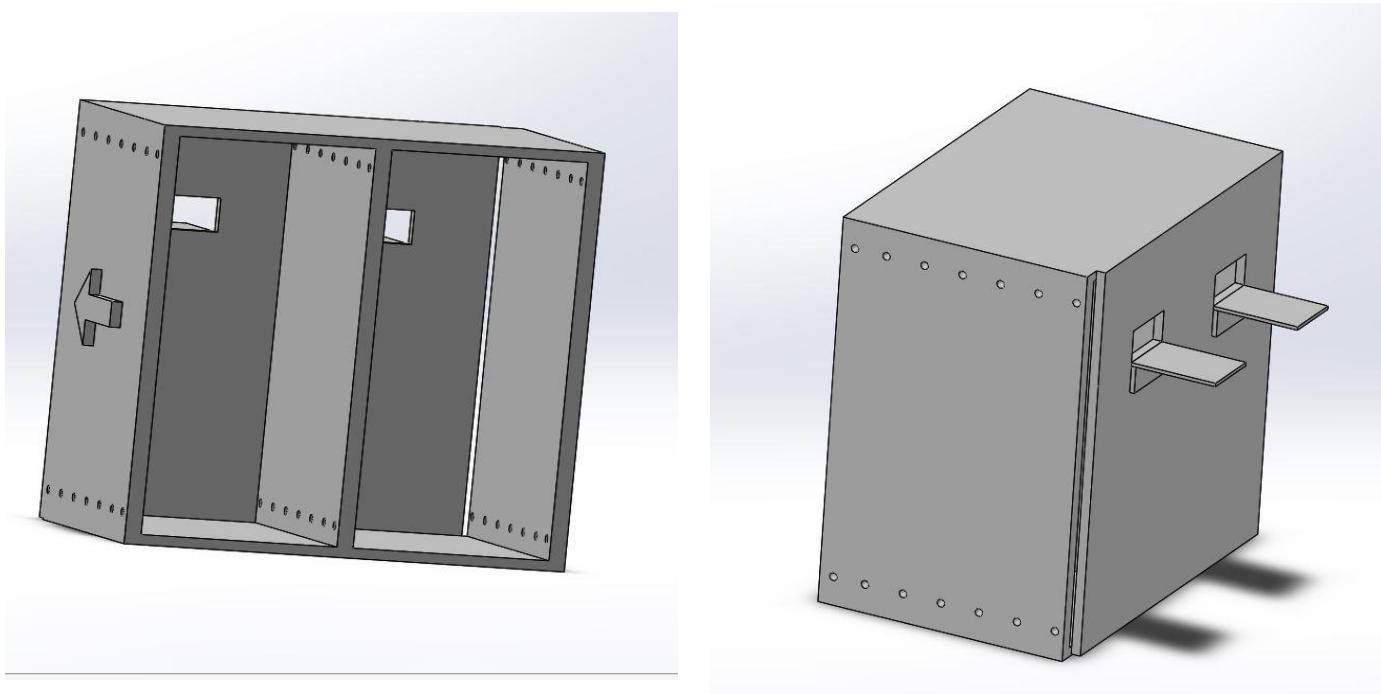


Figure 3.14 Inner frame CAD design

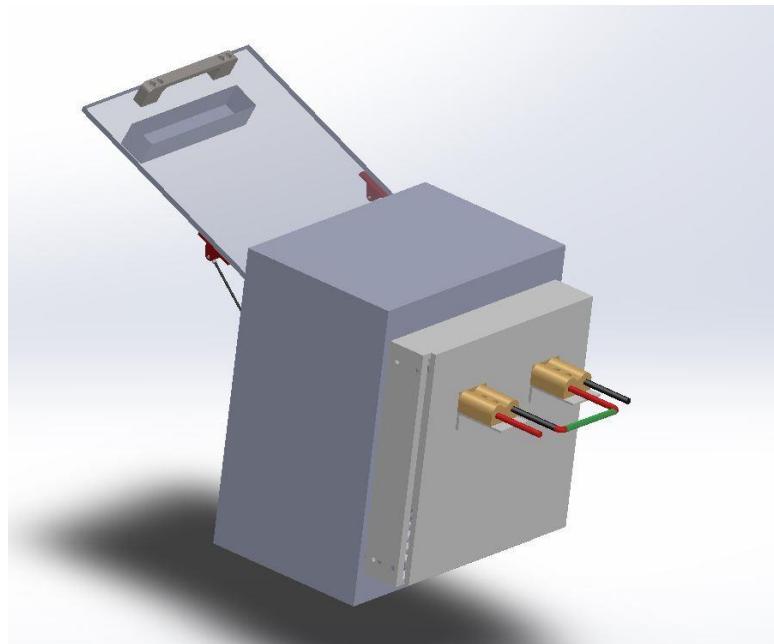


Figure 3.15 Rear view of the inner frame with Anderson connection

3.5 FUTURE MODIFICATIONS

After the prototype has been built, another modification to the design was made. A chamber for supercapacitor modules was added to the battery swapping mechanism, so that these super capacitor modules can be removed from the electric vehicle if needed. This also allows the all the power source and related controllers to be lumped into a single unit. Super capacitor module was designed to match the already available supercapacitor "Maxwell BMOD0058 E016 B02"

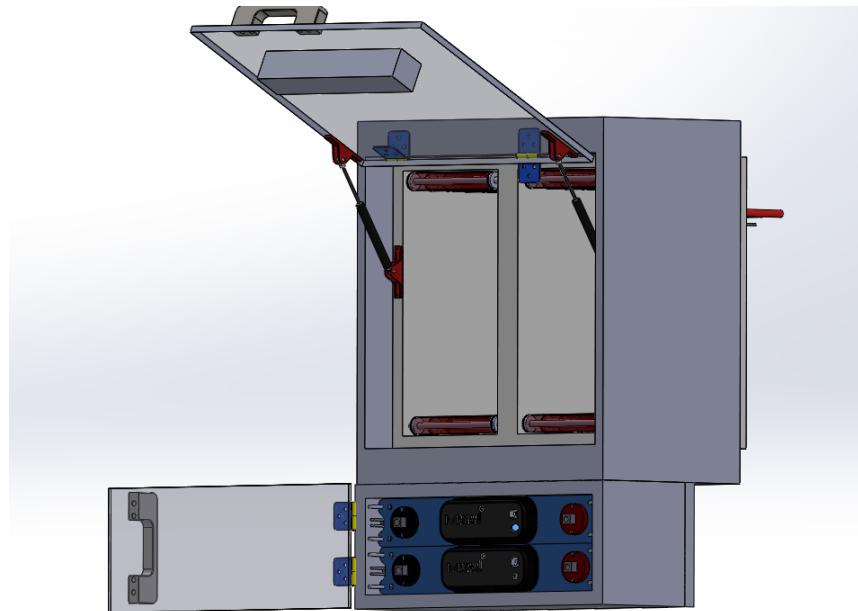


Figure 3.16 Swapping mechanism with super capacitor module added

CHAPTER 4 PROTOTYPE IMPLEMENTATION

The prototype is implemented in the engineering workshop and a preliminary market research was performed to find the best available materials for this design implementation.

4.1 MATERIAL SELECTION

The battery model consists of a battery case that can withstand the weight of the battery and has a handle that is used to eject the battery, as shown in figure 3.3. For this implementation sealed type calcium batteries were used, so GI sheet was used for the outer casing (18Gauge) with electronic isolation between the battery and the casing provided by a removable battery clamp as shown in figure 4.1. The wiring of the battery is done through 0AWG Wire to ensure the current rating meets the requirements. The AWG wire current rating is shown in Table 4.1. For the isolation of the battery from the metallic parts, carbon fiber layer was used.

The electrical connection between the battery and the slider mechanism is achieved by using Anderson connectors which can be handled up to 175A of current which is shown in figure 4.2. The Anderson connector attached to the batteries is shown in figure 4.3

Table 4.1 Gauge chart for the copper wires

Size, AWG	A(RMS), (90°C wire)	Ohms/100 Ft, (One Way)	Voltage Drop/100 Ft, (Column 2 x Column 3)
18	14	0.639	8.95
16	18	0.402	7.24
14	25	0.253	6.33
12	30	0.159	4.77
10	40	0.100	4.00
8	55	0.063	3.47
6	75	0.040	3.00
4	95	0.025	2.38
3	115	0.020	2.30
2	130	0.016	2.08
1	145	0.012	1.74
0	170	0.0098	1.67
00	195	0.0078	1.52
000	225	0.0062	1.40
0000	260	0.0049	1.27



Figure 4.1 Removable battery clamps with isolation



Figure 4.2 Anderson connectors

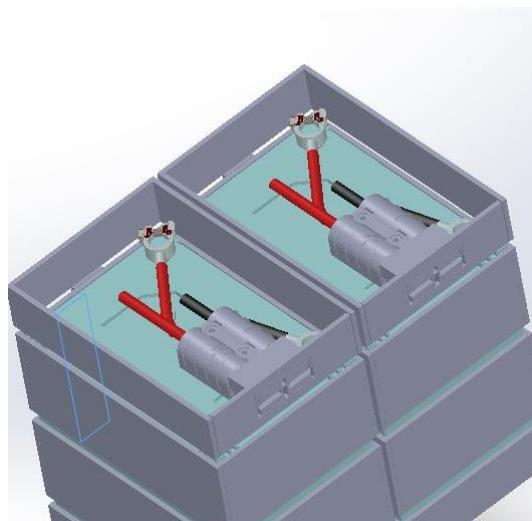


Figure 4.3 Anderson connected to the casing with the wiring attached

4.1.1 Bill of Quantities (BOQ)

After performing the preliminary market research, the BOQ was prepared as shown in the Table 4.2. Some materials such as gas spring, Anderson connectors and isolation battery clamps were ordered from china considering the unavailability in the local market.

Table 4.2 Bill of Quantities for the prototype

ITEM NO	SWAPPER PART	ITEM	Quantity	Remarks	COST
	BATTERY CASE				
01_a		GI SHEET (18GAUGE)	8' X 4' SHEET	Zinc coated	2800.00
02_a		Hinges & Handles			1385.00
03_a		ANDERSON CONNECTOR (175A)	4x	\$18.46 http://bit.ly/2KOqT Gh	3363.34
04_a		BATTERY CLAMP	4X	\$9.66 http://bit.ly/31FL6nL	1757.42
05_a		Flat iron		¾"	380.00
	SLIDER BARS				
06_a		SS RODS 10mm	21.5'	Only for the battery swapper on vehicle	4275.00 687.00
07_a		BALL BEARING 609_9MM_23MM	56X	Dee & dee shop	2240.00
08_a		GI pipe 3/4"	18'		1300.00
	OUTER CASE WINDOW				
09_a		GAS SHOCK ABSORBER	2X	http://bit.ly/2XriLSI	1216.21
10_a		Auto Paint			2450.00
	INNER CHAMBER				
11_a		HEAVY DUTY BOX BAR ¾ x ¾ x1.4mm	18'	GI	900.00
				SUB TOTAL	23203.97

4.2 ENGINEERING DRAWINGS

Engineering drawings for each of the parts were designed so that the technical officers at the engineering workshop could build the prototype according to the specifications needed. The engineering drawings are shown in figure 4.4,4.5,4.6 and 4.7 below.

The dimensions of the design are chosen such that it can be fitted into already available electric e-wheeler on the department.

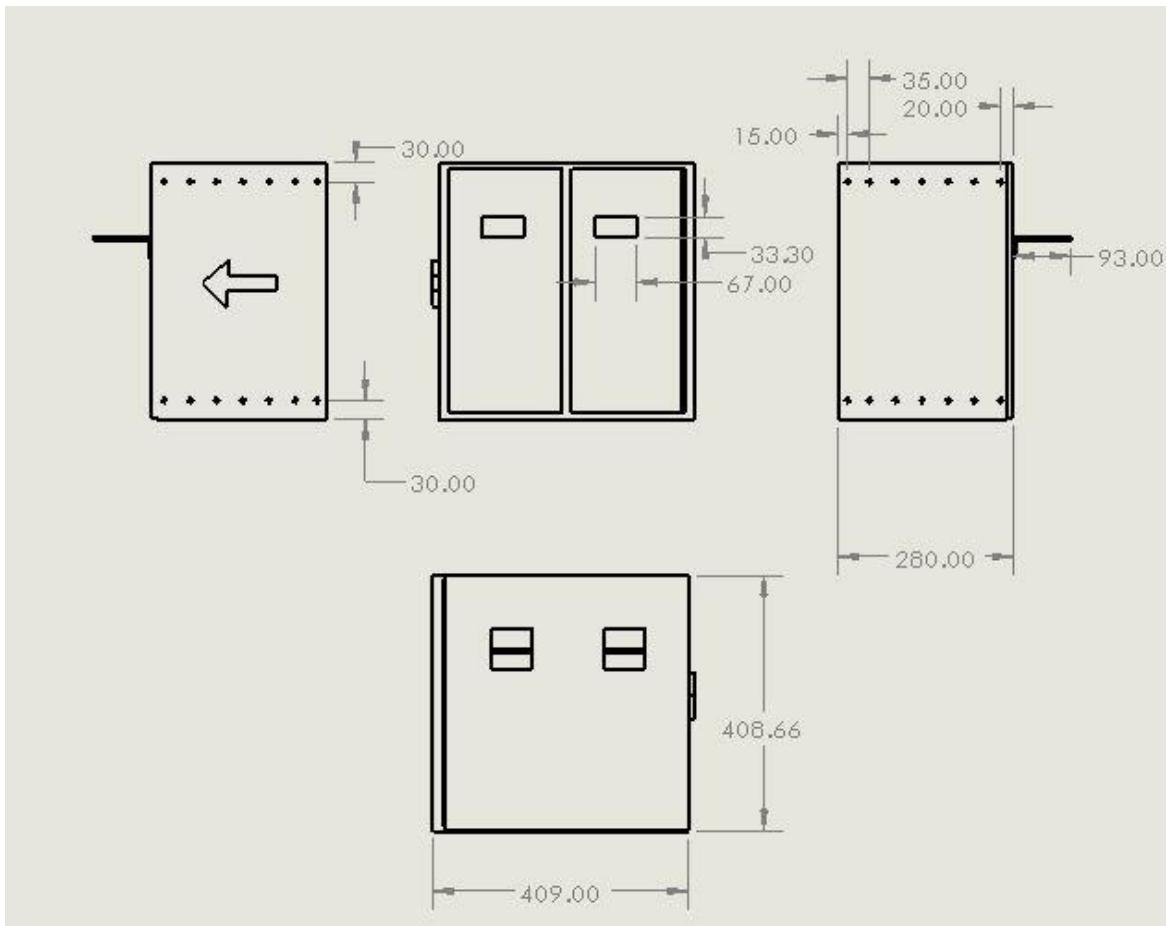


Figure 4.4 Engineering drawing for the inner frame

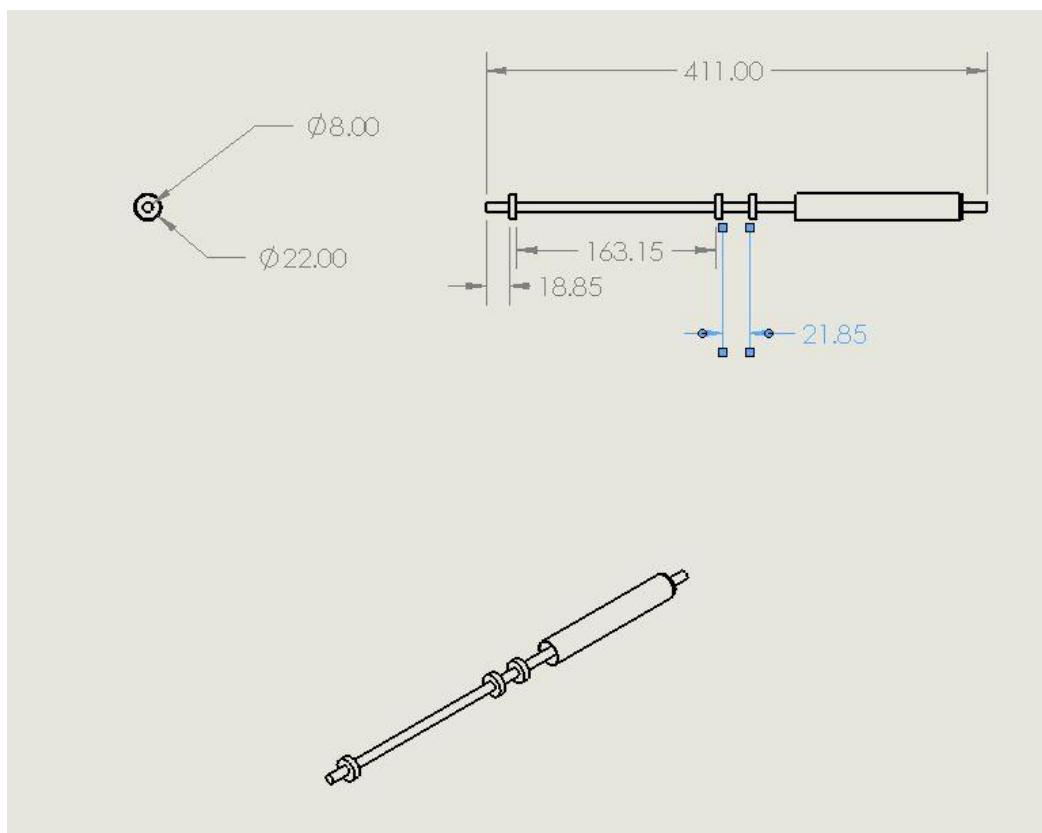


Figure 4.5 Engineering drawing for the sliders

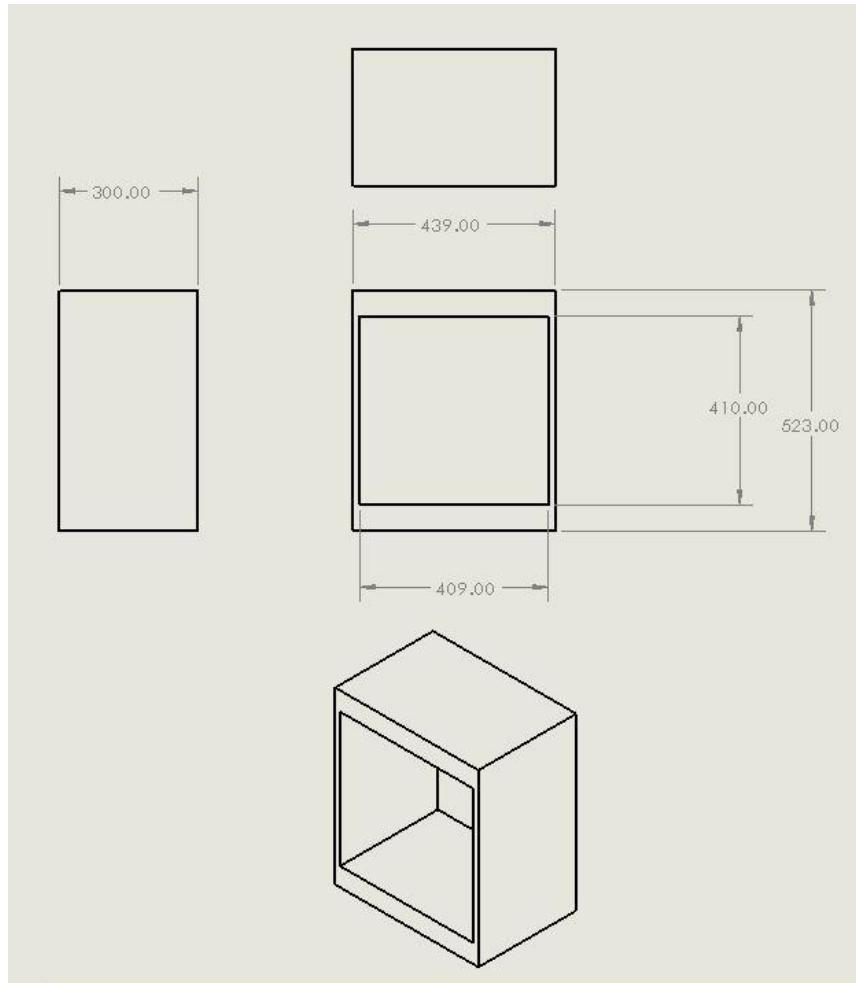


Figure 4.6 Engineering drawing for the outer case

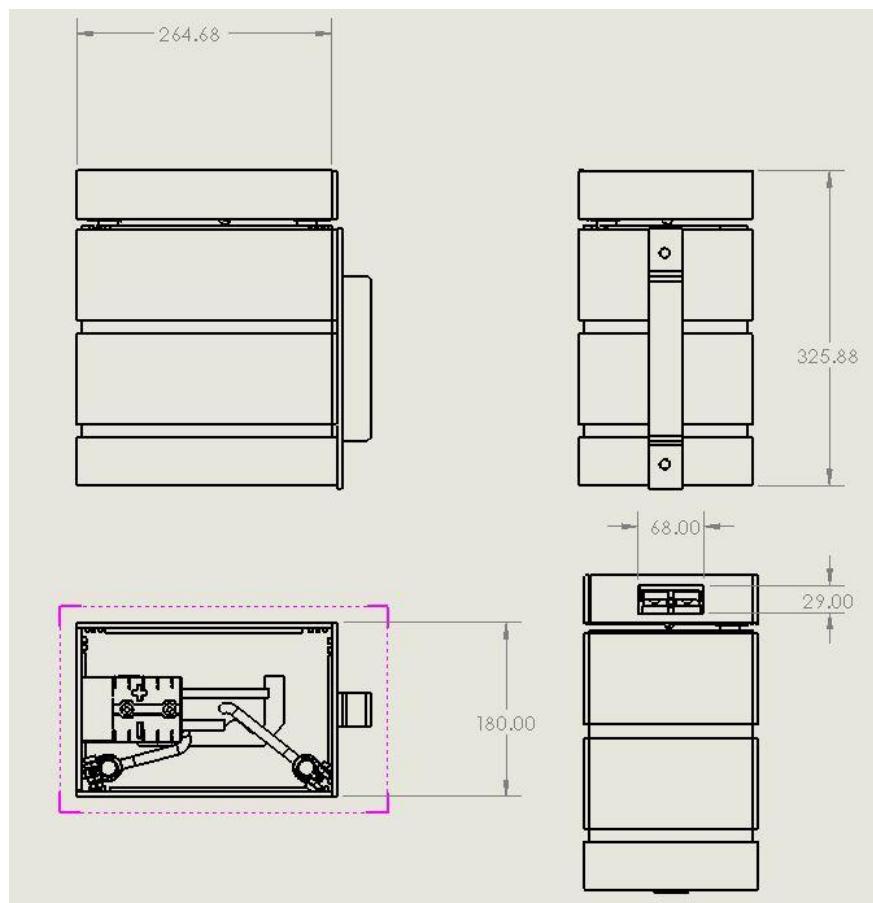


Figure 4.7 Engineering drawing for the battery case

4.3 WORKSHOP PROGRESS

The progress at the engineering workshop is shown in the figures below.



Figure 4.8 Engineering workshop progress I



Figure 4.9 Engineering workshop progress II



Figure 4.10 Engineering workshop progress III



Figure 4.11 Engineering workshop progress iv



Figure 4.12 Engineering workshop progress v



Figure 4.13 Engineering workshop progress vi



Figure 4.14 Engineering workshop progress vii



Figure 4.15 Engineering workshop progress viii



Figure 4.16 Engineering workshop progress ix

CHAPTER 5 BATTERY STATE ESTIMATOR

5.1 INTRODUCTION

One of the tasks of this project was to integrate the battery state estimator that is based on the extended Kalman filter into Android platform, so a summary of already implemented Extended Kalman filter (EKF) based battery estimator is included in this chapter.

5.2 KALMAN FILTER

The Kalman filter is a mature and advanced method to filter the measurement noise to get an optimal estimation of a dynamic system's state. If the system is nonlinear, a linearization process at each time step will be necessary to approximate the non-linear system. The extended Kalman filter (EKF) will play a great role in these systems. The Extended Kalman Filter is adopted to correct the proposed battery model, and the general diagram can be seen in Figure 5.1. Based on the error between the model estimated voltage and the measured voltage, the EKF adjusts the SOC to change the model output voltage to minimize the voltage error. After a certain number of iterations, this error will converge to zero, while the SOC will converge to its optimal value.

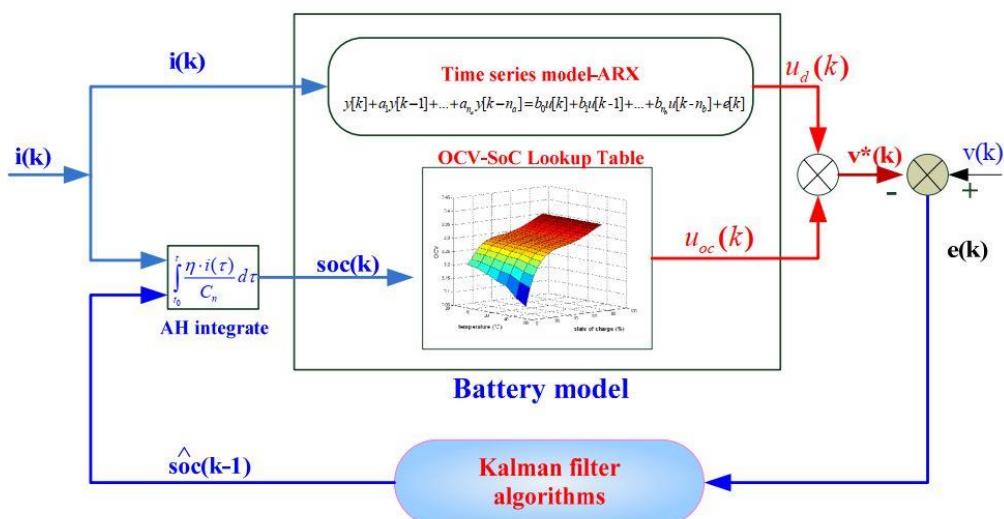


Figure 5.1 Diagram of EKF based SOC estimation algorithm

5.2 REVIEW OF EXTENDED KALMAN FILTER

The Kalman Filter is an optimum state estimator for linear systems that is used to intelligently estimate the states of a dynamic system. The same procedure is adopted by the EKF, having the difference that is implemented for non-linear systems. The nonlinear dynamic systems are linearized at every time step by a predetermined process, in order to be approximated by a linear time varying system.

5.3 KALMAN FILTER FOR SOC ESTIMATION

The Kalman filter is a mature and advanced method to filter the measurement noise to get an optimal estimation of a dynamic system's state. If the system is nonlinear, a linearization process at each time step will be necessary to approximate the non-linear system. The extended Kalman filter (EKF) will play a great role in these systems. The Extended Kalman Filter is adopted to correct the proposed battery model, and the general diagram can be seen in Figure 2. Based on the error between the model estimated voltage and the measured voltage, the EKF adjusts the SOC to change the model output voltage to minimize the voltage error. After a certain number of iterations, this error will converge to zero, while the SOC will converge to its optimal value. The whole operating principle of the model and EKF will be stated in detail here.

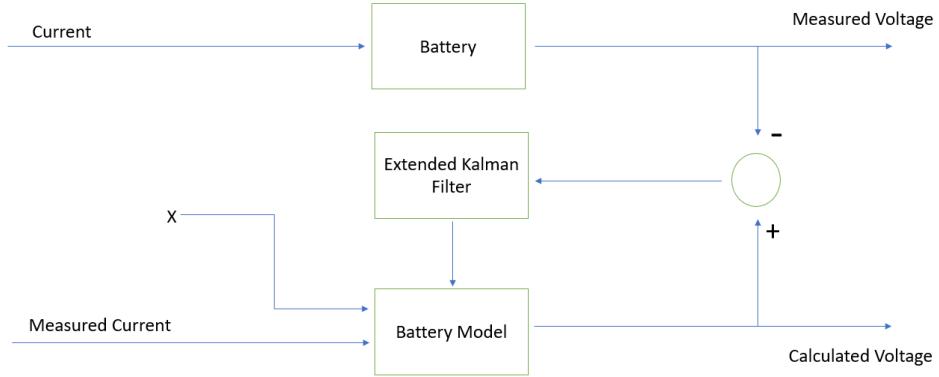


Figure 5.2 Block diagram of EKF

5.4 SOC ESTIMATION BASED ON EKF

The Extended Kalman Filter (EKF) is a method for system state estimation in real time. To estimate the SOC during charging and discharging, the EKF can be constructed as shown in the following steps.

The state space representation can be shortly expressed as:

$$\begin{aligned} x_{k+1} &= f(x_k, i_k) + w_k \\ y_k &= h(x_k, i_k) + v_k \\ w_k &\sim (0, Q_k = \text{diag}(Q_{sk}, Q_{udk}, Q_{udk})) , v_k \sim (0, R_k) \end{aligned}$$

Where x_k is the $[s(k) \ u_d(k) \ u_d(k-1)]$, k is the time index, $h()$ is the output equation of the battery model, w_k is a discrete time process white noise with a covariance matrix $Q_k = \text{diag}(Q_{sk}, Q_{udk}, Q_{udk})$, and similarly v_k is a discrete time observation white noise with covariance matrix R_k . As for the Q_k , the Q_{sk} , Q_{udk} are the covariance of SOC and dynamic voltage u_d respectively. Compute the following partial derivative matrices

$$\hat{A}_k = \left. \frac{\partial f(x_k, i_k)}{\partial x_k} \right|_{x_k = \hat{x}_k^+}, \quad \hat{C}_k = \left. \frac{\partial h(x_k, i_k)}{\partial x_k} \right|_{x_k = \hat{x}_k^-}$$

The initialization can be given by:

$$\begin{aligned} \text{for } k = 0, \text{ set} \\ \hat{x}_0^+ &= E[x_0] = x_0 \\ P_0^+ &= E[(x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^T] = P_{x_0} \end{aligned}$$

Where P_0^+ is the prediction error covariance matrix.

For each of the iterations ($k = 1, 2, \dots$) the following steps are performed:

- Step 1: perform the time update of the state estimate and estimation error covariance:

$$\text{State estimation time update : } \hat{x}_k^- = f(\hat{x}_{k-1}^+, i_{k-1})$$

$$\text{Error covariance time update : } P_k^- = \hat{A}_{k-1} P_{k-1}^+ \hat{A}_{k-1}^T + Q_k$$

- Step 2: compute the Kalman gain matrix:

$$\text{Kalman gain matrix : } L_k = P_k^- \hat{C}_k^T [\hat{C}_k P_k^- \hat{C}_k^T + R_k]^{-1}$$

- Step 3: measurement update:

$$\text{State estimation measurement update: } \hat{x}_k^+ = \hat{x}_k^- + L_k [y_k - g(\hat{x}_k^-, i_k)]$$

$$\text{Error covariance measurement update: } P_k^+ = (I - L_k \hat{C}_k) P_k^-$$

The general diagram of the EKF can be illustrated by Figure 3. After a certain number of iterations, the estimated voltage will converge to the measured value while the estimated SOC will be near to the actual value.

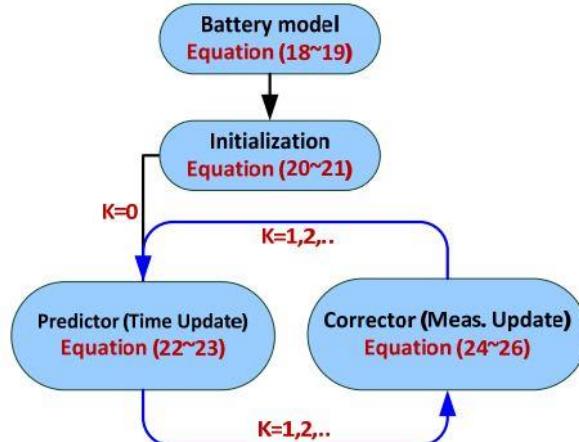


Figure 5.3 General diagram of the extended Kalman filter

5.5 STATE OF HEALTH (SOH)

State of health (SoH) is a figure of merit of the condition of a battery (or a cell, or a battery pack), compared to its ideal conditions. The units of SoH are percentage points (100% = the battery's conditions match the battery's specifications).

Typically, a battery's SoH will be 100% at the time of manufacture and will decrease over time and use. However, a battery's performance at the time of manufacture may not meet its specifications, in which case its initial SoH will be less than 100%.

5.5.1 Parameters

As SoH does not correspond to a particular physical quality, there is no consensus in the industry on how SoH should be determined. The designer of a battery management system may use any of the following parameters (singly or in combination) to derive an arbitrary value for the SoH.

- Internal resistance / impedance / conductance
- Capacity
- Voltage
- Self-discharge
- Ability to accept a charge
- Number of charge-discharge cycles

In addition, the designer of the battery management system defines an arbitrary weight for each of the parameter's contribution to the SoH value. The definition of how SoH is evaluated can be a trade secret.

CHAPTER 6 MOBILE APPLICATION INTEGRATION

6.1 INTRODUCTION

Smart phones with higher processing power are becoming more and more common in the modern world. So, it is important for an IoT platform to integrate its applications into smartphones to provide more accessibility to its users.

So, in this part of this project there were two main objectives.

- Build android application with user interface for the users
- Integrate an extended Kalman filter based battery state estimator into the mobile application

6.2 MOBILE APPLICATION FOR USERS

With a developed mobile application, the battery swapping station can monitor all the electrical vehicles around the battery swapping station (BSS). This is useful for the demand prediction which is to be used on the battery swapping station. Also, the battery swapping station can alert the users about the availability of fully charged batteries in the battery swapping station.

From the user's point of view, the app can be used to determine the state of charge, reserve batteries in the BSS and to get various notifications from the BSS.

6.3 MOBILE APPLICATION FOR BATTERY MONITORING

The main reasons for the mobile phone integration of the battery monitoring are as follows,

- To replace the Raspberry Pi (which was used to estimate the battery parameters using EKF) of the vehicle with our mobile phone.
- Reduce the cost of implementation by replacing the raspberry pi computer
- Better accessibility (communication with the BSS)
- User can make decision regarding vehicle battery by observing the state of the battery.

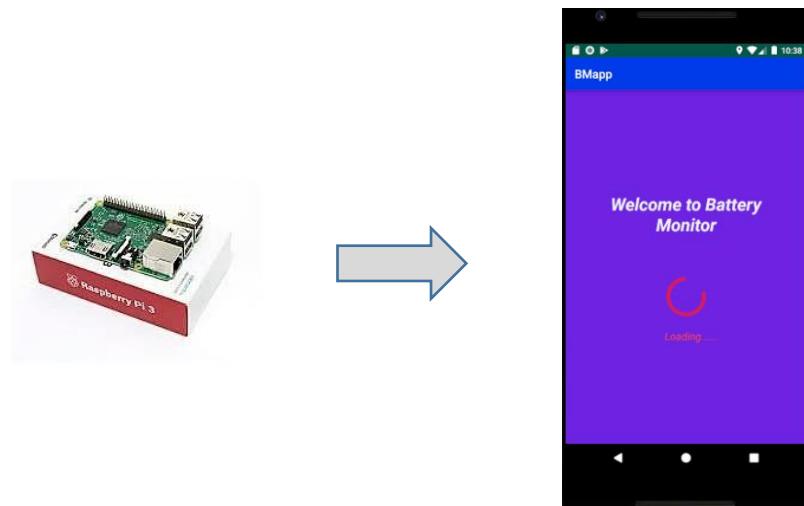


Figure 6.1 Integration of the raspberry pi-based environment into android

6.2.1 Android Studio

Android studio is an integrated development environment (IDE) which comprises of basic tools that is needed when building android based applications. This studio also provides the ability to debug and simulate a android smart device to test the features of the application. Android studio is primarily based on Java and Kotlin programming languages. The main advantage of the android studio is the free availability and the popularity among the developer community, making it easier to debug an application built by android studio

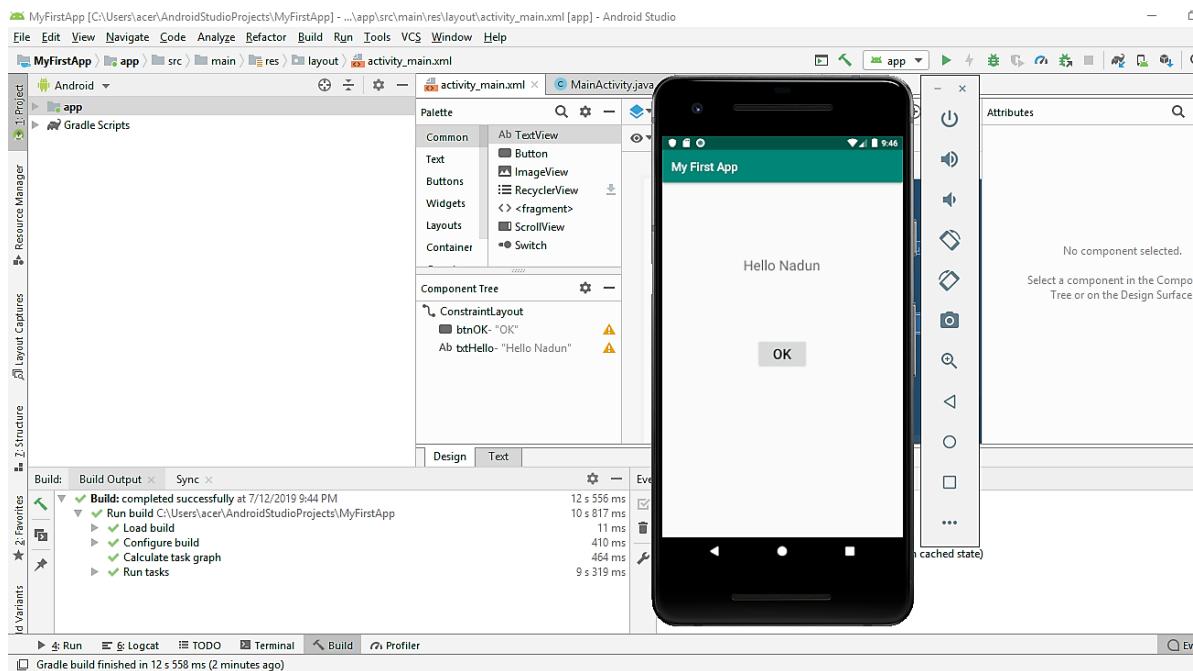


Figure 6.2 Android studio interface

6.2.2 Google APIs

Application programming interface (API) is an interface that allows the client-side application to communicate with a server-side application, to interchange information. Google API allows the integration of google services into third party applications. For example, third party application can use google maps for a unique task that is required by that application. (example- taxi service offered by Uber)

In our project google API is used for implementing the Google Map on the mobile application to find the nearest or desired swapping stations online.

6.2.3 Interface of the application

Present value of voltage and current values of the vehicle battery are received via Bluetooth module to the mobile phone.



Figure 6.3 Bluetooth interface of the android application

The android application is capable of acquiring voltage and current levels used by the ev via Bluetooth communication channel between the EV and the smart phone of the user. The EKF based battery estimator implemented in the android application will estimate SOH , SOC and the temperature based on the voltage and current readings. Screenshot of the application with SOC and SOH levels is shown in figure 6.4.

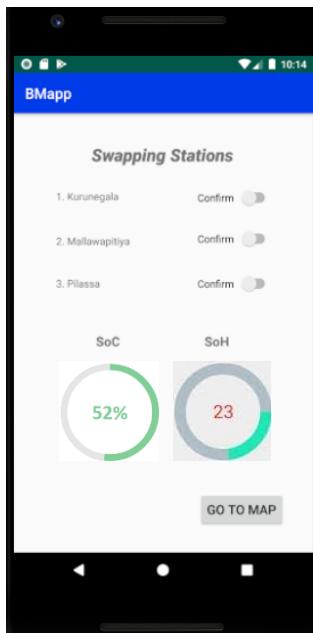


Figure 6.4 Interface of the SoC and SoH display

If the user wants to swap the battery, he can choose a desired swapping station and let the station know that he/she is willing to come. User can see his current location and nearest swapping stations using the map view. If the user is inside the range of a swapping station, the station will be able to see the SoC and SoH levels of the customer vehicle.

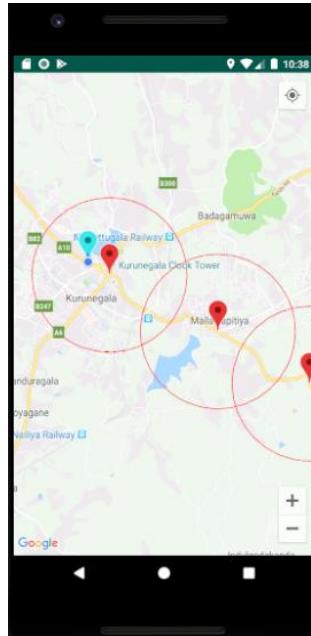


Figure 6.5 Interface of the swapping station display

6.3 JAVA CODE FOR EXTENDED KALMAN FILTER (EKF) ALGORITHM

```

//#initial values assigned
double I0 = 0; //#Input current
double soh = 0.8;
double soc = 1; //#Assumed full charged

double time =0;
double SOH =0;
double SOC =0;
double Vm =0;
double Vp =0;
double Im =0;

double sampleTime =0.2;

//# 3 states declaration

double soc1 = soc;
double soh1 = soh;
double Vth1 = 0;

double[][] vals1 = {{0.05 , 0 , 0},{0, 0.04, 0},{0, 0, 1}};
Matrix Qk = new Matrix(vals1);

double[][] vals2 = {{0.1,0,0},{0,0.1,0},{0,0,0.1}};
Matrix Pkint = new Matrix(vals2);

Matrix Pkin = Pkint;

double[][] vals7 = {{0.5}};
Matrix R1 = new Matrix(vals7); //#(1,

```

```

public void readCsv(){
    BufferedReader br = null;
    String line = "";
    String cvsSplitBy = ",";

    try {
        InputStream is = this.getResources().openRawResource(R.raw.datafile);
        br = new BufferedReader(new InputStreamReader(is));
        while ((line = br.readLine()) != null) {

            // use comma as separator

            String[] values = line.split(cvsSplitBy);
            double t = Double.parseDouble(values[1]);
            I0 = Double.parseDouble(values[2]);
            double V0 = Double.parseDouble(values[3]);

            double r2= 2.5968*(soc*soc*soc*soc)-3.5211*(soc*soc*soc)+ 0*(soc*soc)+1.4757*(soc)-
0.3317;

            double Vth = I0 * r2;
            Vth1 = Vth;

            double r2y = 2.5968*(soc*soc*soc*soc)-
3.5211*(soc*soc*soc)+0*(soc*soc)+1.4757*(soc)-0.3317;
            Vth1 = I0 * r2y;

            double[][] vals3 = {{soh1},{soc1},{Vth1}};
            Matrix y = new Matrix(vals3);

            Matrix yPredict = (Matrix) y.plus(this.statediff(y,I0*0.2));
            soh1 = yPredict.get(0,0);
            soc1 = yPredict.get(1,0);
            Vth1 = yPredict.get(2,0);

            double Voc1=9.8958*(soc1*soc1*soc1*soc1)-
19.3750*(soc1*soc1*soc1)+15.1042*(soc1*soc1)-3.9250*(soc1)+24.06;
            double r01 = 0.0002;
            double voltage = V0;

            double[][] vals4 = {{soh},{soc},{Vth}};
            Matrix x = new Matrix(vals4);

            Matrix xPredict = x.plus(this.statediff(x,I0*sampleTime));
            soh= xPredict.get(0,0);
            soc= xPredict.get(1,0);
            Vth= xPredict.get(2,0);

            double r0 = 0.0002;
            c2 = 200000;
            double Voc=9.8958*(soc*soc*soc*soc)-9.3750*(soc*soc*soc)+15.1042*(soc*soc)-
3.9250*(soc) + 24.06;

            double HKpredict = Voc-Vth-(I0*r0);

            double[][] vals5 = {{1,0,0},{0,1,0},{0,0,(1/(r2*c2))}};
            Matrix Fk = new Matrix(vals5); //#constant here (3,3)

            double[][] vals6 = {{0,0,-1}};
            Matrix HK = new Matrix(vals6); //#constant here (3,3)
            Matrix HKT = HK.transpose();
            Matrix FkT=Fk.transpose();

            Matrix Pk = Fk.times(Pkin.times(FkT)).plus(Qk);
            Matrix Gk =
Pk.times(HKT.times((HK.times(Pk.times(HKT)).plus(R1)).inverse())));
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}

```

```

        Matrix xest=xPredict.plus(Gk.times(voltage-HKpredict));

        Matrix PK= (identity(3,3).minus(Gk.times(HK))).times(Pk);
        soh=xest.get(0,0);
        soc=xest.get(1,0);
        Vth=xest.get(2,0);

        double vest=Voc-Vth-I0*r0;
        Qk = Pkin;
        Pkin = PK;

        time = Double.parseDouble(values[1]);
        Vm = V0;
        Im = I0;
        SOH = soh;
        SOC = soc;
        vp = HKpredict;

        System.out.println(time);
        System.out.println("Vm Measured Voltage "+Vm);
        System.out.println("Im Measured Current "+Im);
        System.out.println("soh "+SOH);
        System.out.println("soc "+SOC);
        System.out.println("Vp Predicted Voltage "+vp);

        testString = Double.toString(SOH);
        txt.setText("SOH : " +testString);

        testString2 = Double.toString(SOC);
        txt2.setText("SOC : " +testString2);

        testString3 = Double.toString(Vm);
        txt3.setText("Vm Measured Voltage : " +testString3);

        testString4 = Double.toString(Im);
        txt4.setText("Im Measured Current : " +testString4);

    }

}

public Matrix statediff(Matrix x, double u) {

    double soh = x.get(0,0);
    double soc = x.get(1,0);
    double vc = x.get(2,0);

    double current = u;
    double r2 = 2.5968*(soc*soc*soc*soc)-3.5211*(soc*soc*soc)+0*(soc*soc)+1.4757*(soc)- 0.3317;
    double c2=200000;

    double CAPACITY_NEW = 20;
    double CAPACITY_OLD_PERCENT = 80;

    double alpha = CAPACITY_OLD_PERCENT /100;
    double beta = 1 - alpha;
    double capacity = CAPACITY_NEW * (alpha + beta*soh);

    double lifetime = 2000;

    double dsoh = -Math.abs(current) / (2*lifetime*capacity*3600);
    double dsoc = - current / (capacity*3600);
    double dvc = (-vc) / (c2*r2) + (current/c2);

    double[][] vals1 = {{dsoh},{dsoc},{dvc}};
    return new Matrix(vals1);

}

```

6.4 CONCLUSION

By creating a mobile application, it minimizes the resources (In this case the raspberry pi was initially used to estimate the battery status using extended Kalman filter), time and cost for the Battery monitoring System in a vehicle and it makes easier to the user to monitor the vehicle battery state and make decisions regarding the battery at the moment or in near future.

It is also possible for implement additional features such as battery reserving, billing service-based account management, battery delivery services to a location using this mobile app integration.

CHAPTER 7 DEMAND PREDICTION OF A BATTERY SWAPPING STATION USING TIME SERIES DATA STATISTICS

7.1 SELECTION OF A SUITABLE DISCRETE DISTRIBUTION MODEL

To select the most appropriate model for the forecasting of the future demand for the batteries, a model needs to be selected that has the highest precision when comes to prediction. The prediction needs to be done for all the one-hour slots for a given day.

7.1.1 Normal Distribution

For a normal distribution the probability density function (PDF) is defined as

$$f(x) = \frac{1}{\sqrt{2\pi s^2}} \exp \left[-\frac{(x-x')^2}{2s^2} \right]$$

By using the maximum likelihood estimation (MLE) the mean and the variance for the normal distribution is given by the following equations

$$\bar{X} = \sum_{i=1}^n x_i / n$$

$$s^2_{MLE} = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$$

7.1.2 Log normal distribution

For a log normal distribution, the PDF is defined as

$$f(x) = \frac{1}{x\sqrt{2\pi s^2}} \exp \left[-\frac{(\ln(x) - \mu)^2}{2s^2} \right]$$

Where \bar{X} and s^2 are the scale and shape parameters of the distribution. The corresponding estimation for the mean and variance is given by,

$$\begin{aligned} \bar{X} &= \sum_{i=1}^n \ln(x_i) / n \\ s^2_{MLE} &= \frac{1}{n} \sum_{i=1}^n [\ln(x_i) - \ln(\bar{x})]^2 \end{aligned}$$

7.1.3 Weibull Distribution

The PDF for the distribution is given as

$$f(x) = \frac{\lambda}{\sigma} \left(\frac{x}{\sigma} \right)^{\lambda-1} \exp \left[-\left(\frac{x}{\sigma} \right)^\lambda \right] \text{ for } x \geq 0$$

Where λ and σ are scale and shape parameters for the distribution.

7.1.4 Gamma distribution

The PDF for the beta distribution is given as,

$$f(x) = \frac{x^{\lambda-1}}{\sigma^\lambda \tau(\lambda)} \exp\left(-\frac{x}{\sigma}\right) \text{ for } x \geq 0$$

Where σ and λ are scale and shape parameters of the distribution. τ is the gamma function.

7.1.5 Log logistic distribution

The PDF of the distribution is given by

$$f(x) = \frac{1}{\sigma} \frac{\left[\exp \frac{\ln(x)}{\sigma}\right]}{\left[1 + \exp \frac{\ln x}{\sigma}\right]^2}$$

Where σ is the scaling parameter.

7.2 METHODOLOGY

Several methods have been tested to determine the optimum model to predict the battery demand of the BSS.

7.2.1 Simple Moving average technique

SMA models time series data from the observations from previous days on that specific hour. And the averaged mean is used to predict the demand of the batteries in the BSS. The SMA is given by,

$$SMA(n) = \frac{1}{n} \sum_{t=k-n+1}^k A_t$$

Where the n is the span of the data sets being considered. And k is the relative position of the period currently being considered within the n , and A_t is the actual value for the time t .

➤ Error forecasting

Mean square error (MSE) and mean absolute percentage error (MAPE) were used to estimate the goodness of the SMA technique. The MSE of a SMA test can be shown by,[2]

$$MSE = \frac{1}{n} \sum_{t=1}^n (A_t - F_t)^2$$

Where n is the total number of data, A_t is the actual data that is observed and F_t is the forecasted value for the demand in the BSS. The MAPE is defined as

$$MAPE = \left(\frac{1}{n} \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right| \right) * 100\%$$

➤ **Goodness of fit tests**

Goodness of fit tests were used to select the most appropriate statistical distribution model to determine the predicted vehicles that are arrive in a specific hour slot of the day. Tests such as Chi square, Kolmogorov-Smirnov, Anderson darling test, Shapiro Wilk, parameter estimation can be used in determine the best suitable model for the statistical prediction

7.2.2 Parameter estimation

For a data set obtained follows a certain selected distribution, then the parameters can be estimated using that data set. There are some commonly used parameter estimators such as Maximum likelihood estimators(MLE)[3], Least Squares Estimation method (LSE),Method of Moments(MoM). The most suitable can be used to determine the statistical distribution that gives the minimum estimation error. The MLE is defined as:

$$L(\theta) = \prod_{i=1}^n f_x(x_i | \theta)$$

Where $L(\theta)$ is the maximum likelihood function and $f_x(x_i)$ is the probability distribution function and x_i are the observed values of in the random sample of size n . θ is the set of unknown parameters. The value which maximizes the likelihood function of ; $f_x(x_i | \theta)$ would be the likelihood estimate θ^* . The maximum likelihood of the θ^* can also be obtained using the maximizing of $\log [f_x(x_i | \theta)]$ function. The MLE maximize the $L(\theta)$ for a the given data set by partial derivatives and equating them to zero.

$$\frac{\partial \ln[L(\theta)]}{\partial \theta_k} = 0$$

➤ **Selection of best distribution model**

Selection of the most suitable distribution model to predict the future demand for the SS is selected by comparing the Root Mean Square error (RMSE) and Mean absolute error (MAE) indices.

➤ **Using the confidence interval**

The confidence interval on the selected data set can be selected once the appropriate distribution model is selected for the prediction. The confidence interval for a normal distribution is given by:

$$P_x = \bar{X} \pm Z_{\alpha/2} S_{\bar{x}}$$

And for the log normal distribution, the percentile is given by

$$P_x = \left(\bar{X} + \frac{S^2}{2} \right) \pm Z_{\alpha/2} \sqrt{\frac{s^2}{n} + \frac{s^4}{2(n-1)}}$$

7.3 DATA SOURCE

For the simulation of the estimators, a time series random data set is generated on hourly average number of vehicles, comprising of 1000 samples. The samples are generated to show the number of electrical vehicles with a certain State of Charge (SOC) range in an hourly basis. The data set is then biased on hourly basis in order to match with daily traffic congestion on a typical weekday. For this purpose, statistical data for the Kandy town in SL was used. Then statistical inferences were obtained from the generated set of data.

Table 7.1. Generated time series data for the Monday

No TIMF	Monday										0
	1 0-20%	2 0-20%	3 0-20%	4 0-20%	5 0-20%	6 0-20%	7 0-20%	8 0-20%	9 0-20%	10 0-20%	
12AM	6	7	23	21	2	9	6	2	1	11	
01AM	12	24	12	10	5	0	11	4	9	18	
02AM	8	8	15	0	15	10	18	14	24	9	
03AM	6	23	17	2	0	18	3	25	13	4	
04AM	11	10	9	6	21	0	3	8	17	19	
05AM	4	1	13	16	23	9	17	13	7	9	
06AM	37	31	28	31	23	38	27	31	32	34	
07AM	36	39	35	27	20	37	33	40	31	36	
08AM	42	47	56	37	53	41	32	59	53	56	
09AM	30	52	52	42	48	40	32	51	43	46	
10AM	35	21	37	39	34	28	25	39	38	25	
11AM	48	58	31	40	35	44	41	42	52	36	
12PM	46	69	48	40	40	42	42	48	48	41	
01PM	55	50	65	43	40	62	55	65	53	63	
02PM	33	43	30	25	36	54	49	37	27	26	
03PM	35	34	33	30	27	40	31	33	33	33	
04PM	52	47	50	52	55	43	56	56	40	49	
05PM	39	26	27	38	37	36	42	40	28	33	
06PM	57	50	73	68	57	79	73	66	79	54	
07PM	58	80	53	60	76	62	61	79	61	72	
08PM	67	73	70	75	69	55	76	60	63	54	
09PM	16	6	21	22	8	13	4	11	14	24	
10PM	1	14	1	18	9	10	4	14	9	8	
11PM	9	5	1	14	19	21	13	9	20	25	

7.4 RESULTS

7.4.1 SMA technique

The SMA on all the time slots for the specific day of the week was calculated and the results are used to predict the battery demand of the BSS.

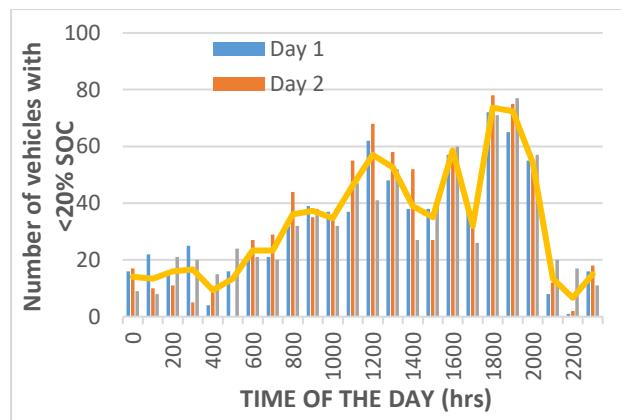


Figure. 7.1. Variation of the SMA with the time on Monday

TABLE 7.2 SMA estimate and the MSE for the generated dataset

Time slot	SMA	MSE	MAPE (%)
1.00pm-2.00pm	55.39	54.6121	15.39%
5.00pm-6.00pm	34.98	36	20.50%

7.4.2 Distribution model technique

The calculated mean and the variance for the data set is shown in the table 7.3

Table 7.3 parameter estimation for the different distributions

DISTRIBUTION TYPE	$\hat{\mu}$		$\hat{\sigma}$	
	For 2-3pm	For 5-6pm	For 2-3pm	For 5-6pm
Normal	55.305	34.9800	9.0304	7.1453
Log normal	4.0019	3.5396	0.1608	0.1757
Weibull	59.1224	37.571	7.3372	6.6498
Gamma	39.5366	33.0808	1.4012	1.0574
Log logistic	55.4940	34.9933	5.2596	3.6457

The PDF corresponding to different distribution types with the estimated mean and variance values substituted is shown in Table 7.4.

Table 7.4 pdf for the different distributions for 2-3pm interval

DISTRIBUTION TYPE	PDF
Normal	$\frac{1}{\sqrt{2\pi(9.0304)^2}} \exp\left[-\frac{(x - 55.305)^2}{2(9.0304)^2}\right]$
Log normal	$\frac{1}{x\sqrt{2\pi(0.1608)^2}} \exp\left[-\frac{(\ln(x) - 4.0019)^2}{2(0.1608)^2}\right]$
Weibull	$f(x) = 8.057\left(\frac{x}{7.3372}\right)^{58.1224} \exp\left[-\left(\frac{x}{7.3372}\right)^{59.1224}\right]$
Gamma	$f(x) = \frac{x^{38.5366}}{619607.95\tau(39.5366)} \exp(-\frac{x}{1.4012}) \text{ for } x \geq 0$
Log logistic	$f(x) = \frac{1}{5.2596} \frac{[\exp \frac{\ln(x)}{5.2596}]}{\left[1 + \exp \frac{\ln(x)}{5.2596}\right]^2}$

The error indices for the distribution models is shown in the table 7.5 below

TABLE 7.5 ERROR INDICES FOR THE DISTRIBUTION MODELS

DISTRIBUTION TYPE	MAE		MPAE (%)	
	1.00-2.00pm	5.00-6.00pm	1.00-2.00pm	5.00-6.00pm
<i>Normal</i>	7.305	13.02	13.2	37.220
<i>Log normal</i>	0.1307	0.3316	3.26	9.3682
<i>Weibull</i>	11.1224	6.429	18.8112	17.11
<i>Gamma</i>	8.4634	10.9192	21.4064	33.0007
<i>Log logistic</i>	7.494	9.0067	8.0981	25.7383

based upon the obtained results the minimum error indices are obtained for the log normal distribution, so the confidence interval for the distribution is given in table 7.6 below.

TABLE 7.6 CONFIDENCER INTERVALS FOR THE DISTRIBUTION MODELS

Confidence interval	Demand
99%	(4.0016, 4.0279)
95%	(4.0047, 4.0248)
90%	(4.0063, 4.0232)

7.5 CONCLUSION

By selecting the appropriate prediction model, the BSS can use that model to predict the demand for future time slots. Then, optimize its energy management and cost optimization scheme for improve the efficiency of the BSS.

CHAPTER 8 DEMAND FORECASTING FOR A BATTERY SWAPPING STATION USING ARIMA MODELLING

8.1 DATA GATHERING

By using the communication IOT platform, the Battery Swapping station (BSS) can be used to collect the average number of vehicles within a given time interval with a specific SOC range. The clustered SOC ranges can be used to determine the demand of the batteries by the BSS

8.2 AUTOREGRESSIVE INTEGRATED MOVING AVERAGE

Typically, a time series data consists of trend, seasonality and random fluctuations. To address this data an ARIMA process is used, which is a mathematical model that can be used in time series forecasting.

The ARIMA(p,d,q) model consists,

- p – the number of autoregressive terms
- d – the number of differencing required
- q – the number of moving averages

and this model assumes that the data set is univariate, meaning the trends and seasonality's needs to be removed before applying the ARIMA model.

To approach this, box-Jenkins method used iterative process which consisted of 3 stages, model identification, parameter estimation, and diagnostic checking. Literature by Makridakis, Wheelwright and Hyndman further extended this into five-stage process

- Data preparation
- Model selection
- Parameter estimation
- Diagnostics
- Forecasting

8.3 METHODOLOGY

8.3.1 Data generation

The ARIMA model requires adequate historical data. For the testing of the proposed method, a random data set was generated hourly with the SOC ranges of 20% each (0-20%, 21-40%...). The data set is biased hourly at off peak times and lunch, office starting hours to make the data set more realistic. Table 8.1 shows the 0-20% SOC data set generated for 8 days, for a typical working day.

Table 8.1. Time series data for 9 consecutive Mondays without a trend

No TIMF	Monday	1	2	3	4	5	6	7	8
		0-20%	0-20%	0-20%	0-20%	0-20%	0-20%	0-20%	0-20%
12AM		13	8	18	1	3	9	23	11
01AM		14	18	3	12	24	14	13	7
02AM		24	9	22	20	22	24	24	14
03AM		14	13	16	14	7	2	17	11
04AM		11	21	13	13	6	25	14	3
05AM		21	8	14	22	16	6	13	8
06AM		21	27	40	35	21	28	31	38
07AM		39	29	36	26	37	40	27	36
08AM		33	48	38	50	36	42	50	45
09AM		41	47	49	31	45	34	48	51
10AM		30	25	29	39	39	27	22	24
11AM		30	53	48	58	59	31	30	32
12PM		45	54	65	45	70	49	58	47
01PM		59	41	61	60	47	66	54	51
02PM		28	36	38	46	34	32	30	35
03PM		36	27	25	40	25	32	32	37
04PM		53	56	46	42	52	42	58	53
05PM		27	43	41	32	29	26	41	43
06PM		71	62	69	60	56	74	80	50
07PM		63	73	80	59	63	78	70	65
08PM		78	64	69	50	78	62	61	54
09PM		17	4	4	17	22	23	7	12
10PM		15	25	0	5	17	8	16	14
11PM		9	2	10	18	3	14	17	8

For the initial part of the testing, the data set is generated without any trend. And for the second part of the test the data set is generated with a upward trend, which is quite possible for a BSS as the number of customers in the area increases gradually. Figure 2 and 3 shows the time domain data without a trend and with a trend respectively. For each of the data set of 24 represents number of vehicles arrived at the BSS for a complete day. The time series data was analyzed and modeled by using MATLAB software package.

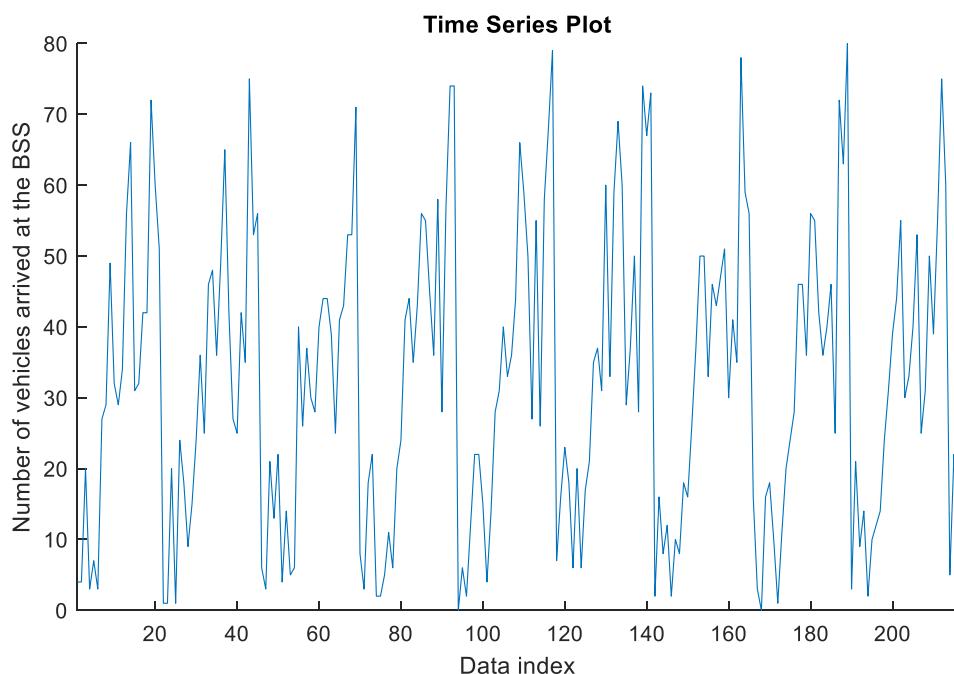


Figure. 8.1. Time series data for 9 consecutive Mondays without a trend

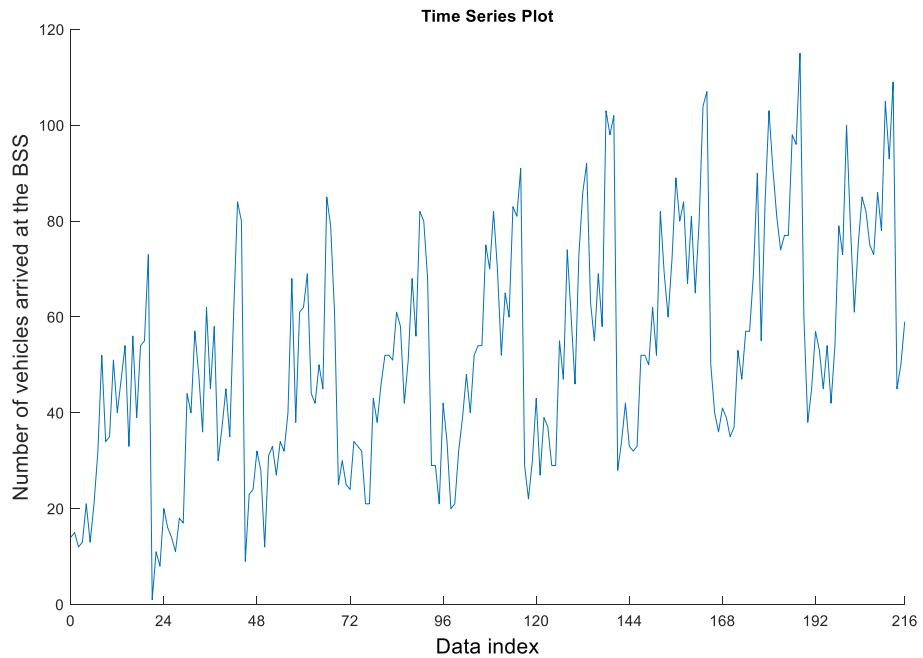


Figure. 8.2. Time series data for 9 consecutive Mondays with a trend

8.3.2 Model constraints

In order to design the modeling of the data set simpler, the following assumptions are made on the battery swapping eco system.

1. The battery must be replaced by the vehicle if SOC level is below 20%
2. The total number of batteries in the BSS is constant, which includes fully charged batteries, drained batteries, and the batteries are under charging.

8.3.3 Data preparation

This step consists of manipulation of data to remove trending or seasonality that is available in the time series data. This also involves differencing of consecutive data or transformations such as mean removal, log transformations depending on the data set.

Upon observation for the both of the cases the data set generated has a 24-hour repeating data set, the seasonality is set to 24. And for the time series data with the trend, the trend was removed under the data preparation as shown in figure 9.3.

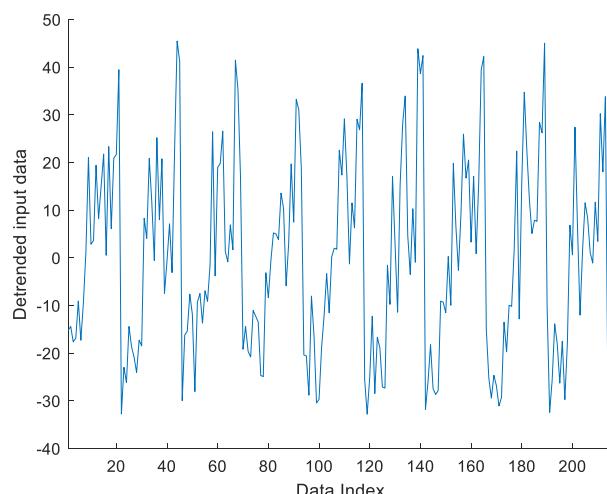


Figure 8.3. Time Series plot with the trending removed

For the assessment of the modified time series data is stationary, Dickey Fuller unit root test was performed on the data set to confirm the stationarity.

H0 – Null hypotheses - series has a unit root

H1 – Series does not have a unit root. The series is stationary

8.3.4 Parameter estimation

For the selection of the p and q parameters for the ARIMA model the Autocorrelation function (ACF) and the partial autocorrelation functions (PACF) were obtained for the both of the cases as shown in figure 9.4 and 9.5 for time series data without a trend as shown below.

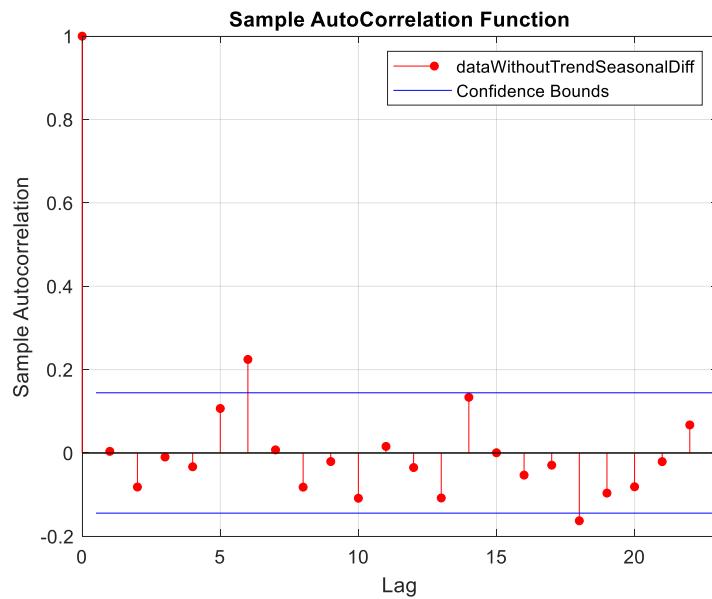


Figure 8.4. ACF Correlograms for the time series data without a trend with 95% confidence intervals

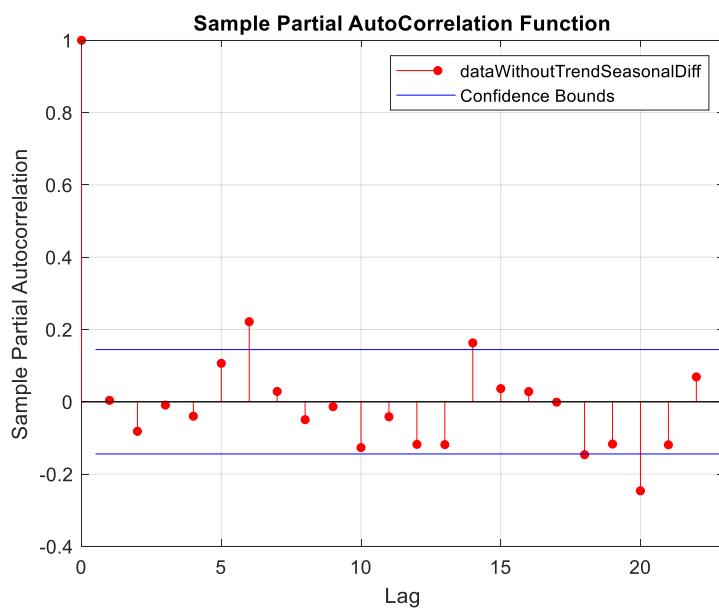


Figure 8.5. ACF Correlograms for the time series data with a trend with 95% confidence intervals

Observing on the ACF and PACF shows that both truncates after lag1, and some noteworthy spikes present at lag 6 of both ACF and PACF. So, the ARIMA (1,0,1) with constant was selected and tested for the time series data.

9.4 MODEL CHECKING

The model equation for the model selected is

$$(1 - \phi_1 L)Y_t = c + (1 + \theta_1 L) \epsilon_t$$

Where

- L - Backshift operator
- ϵ_t - Random noise occurring at time t
- c, θ_1, ϕ_1 - coefficients to determine

From the model estimators the coefficients can be estimated and as shown in TABLE I. The extracted equation for the model is

$$Y_t = \epsilon_t + 0.7107 \epsilon_{t-1} - 0.6745 Y_{t-1} - 0.3550$$

And the estimator and the data set variation are shown in figure 9.6

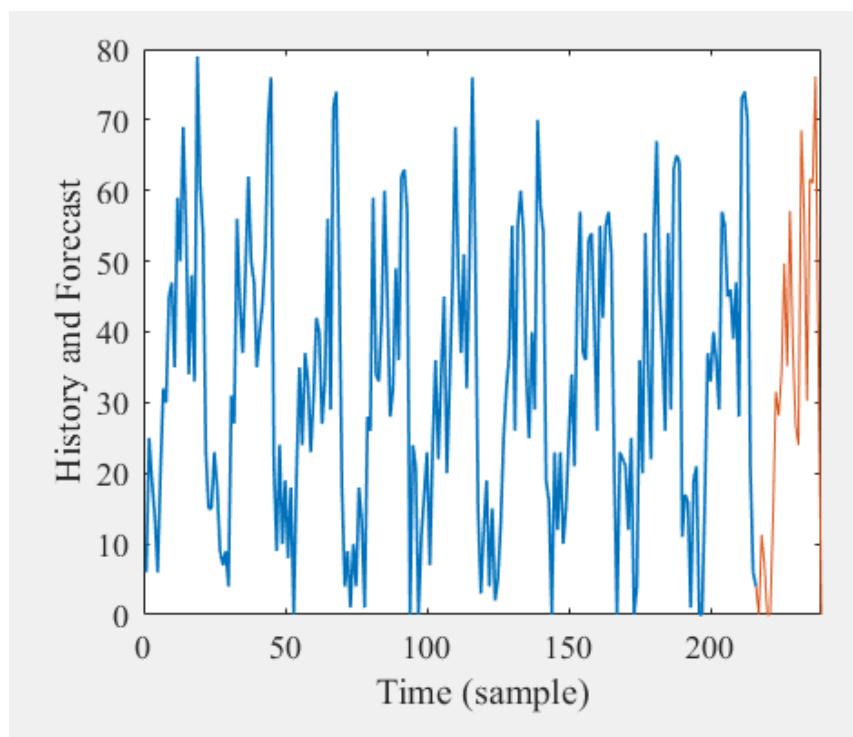


Figure 8.6. Historical data and the forecast for the ARIMA(1,0,1) without trend using one iteration

Similarly, the coefficients were estimated for the historical data with the trend as shown in figure 9.7, tests were performed with other ARIMA coefficients as shown in Table 8.2

TABLE 8.2 ESTIMATED PARAMETERS FOR DIFFERENT MODELS FOR THE TIME SERIES DATA WITHOUT A TREND

	Parameter	Value	StandardError	TStatistic	PValue
ARIMA(1,0,1)	Constant	-0.35501	1.3534	-0.2623	0.79309
	AR{1}	-0.67452	0.57528	-1.1725	0.241
	MA{1}	0.71071	0.54827	1.2963	0.19488
ARIMA(1,0,2)	Constant	-0.2297	0.8596	-0.2672	0.7893
	AR{1}	-0.1162	0.8899	-0.1306	0.8961
	MA{1}	0.1202	0.8866	0.1356	0.8922
	MA{2}	-0.0839	0.0692	-1.2125	0.2253
ARIMA(2,0,2)	Constant	-0.2264	0.8715	-0.2598	0.7950
	AR{1}	-0.1342	0.9448	-0.1420	0.8871
	AR{2}	0.0337	1.0261	0.0328	0.9738
	MA{1}	0.1381	0.9362	0.1475	0.8827
	MA{2}	-0.1163	1.0234	-0.1136	0.9095
ARIMA(1,0,0)	Constant	-0.1971	0.7958	-0.2477	0.8044
	AR{1}	0.0038	0.0756	0.0508	0.9595
ARIMA(0,0,1)	Constant	-0.1979	0.7991	-0.2479	0.8044
	MA{1}	0.0046	0.0755	0.0605	0.9517

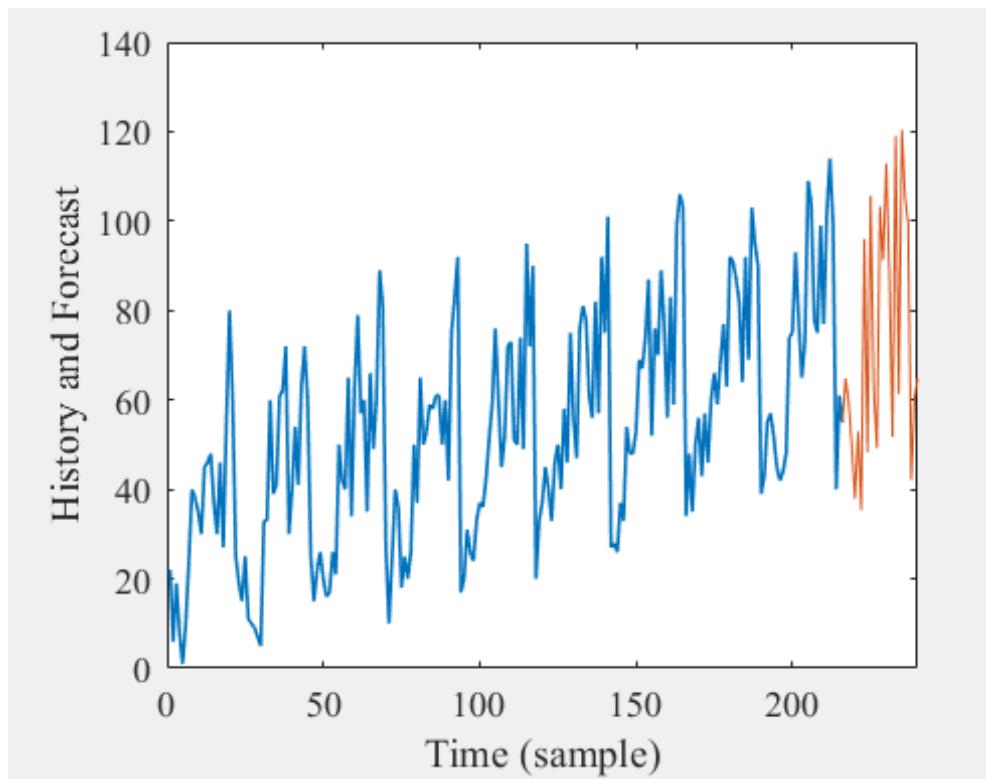


Figure 8.7. Historical data and the forecast for the ARIMA(1,0,1) with trend using one iteration

TABLE 8.3 ESTIMATED PARAMETERS FOR DIFFERENT MODELS FOR THE TIME SERIES DATA WITH A TREND

	Parameter	Value	StandardError	TStatistic	PValue
ARIMA(1,0,1)	Constant	0.34399	0.1283	0.80316	0.42188
	AR{1}	0.92851	0.0910	10.203	0
	MA{1}	-0.88112	0.11021	-7.9952	0
ARIMA(1,0,2)	Constant	0.38573	0.33078	1.1661	0.24357
	AR{1}	0.91667	0.067233	13.634	0
	MA{1}	0.09556	0.0665	-12.315	0
	MA{2}	0.21673	0.068734	3.1532	0.001
ARIMA(2,0,2)	Constant	0.38366	0.32863	1.1675	0.24303
	AR{1}	0.92179	0.24242	3.8025	0.00014
	AR{2}	-0.00478	0.22696	-0.0210	0.98317
	MA{1}	-1.047	0.23324	-4.4889	0
	MA{2}	0.22138	0.21375	1.0357	0.30036
ARIMA(1,0,0)	Constant	3.3302	0.71875	4.6333	0
	AR{1}	-0.0110	0.0707	-0.1557	0.876
ARIMA(0,0,1)	Constant	3.2959	0.65104	5.0625	0
	MA{1}	-0.0126	0.0707	-0.1788	0.858

The goodness of the selected model was tested by using the Akaike information criteria (AIC) and Bayesian information criterion (BIC) criteria as shown in the table 9.4.

TABLE 8.4 GOODNESS OF FIT TESTS

MODEL	AIC	SEB
ARIMA(1,0,1)	1.4736e+03	1.4866e+03
ARIMA(1,0,2)	1.4749e+03	1.4911e+03
ARIMA(2,0,2)	1.4769e+03	1.4964e+03
ARIMA(1,0,0)	1.4722e+03	1.4820e+03
ARIMA(0,0,1)	1.4722e+03	1.4820e+03

The selected model can be then used to predict the upcoming demand (Rounded off) as shown in the table 9.5

TABLE 8.5 FORECASTED DEMAND FOR THE NEXT TIME STEPS

TIME (HRS)	1200	1300	1400	1500	1600	1700	1800	1900
FORECAST	66	54	57	29	65	30	79	80

9.5 CONCLUSION

Based on the selected ARIMA model, the demand can be forecasted for the next time steps by using the historical time series data. Typically, a battery charging time for lead acid batteries is under 12 hours, so the demand forecast for the next 12 hours would be sufficient for the BSS, in order to optimize the chagrining decisions and maximize the profit of the battery swapping station.

The estimation can be made for typical workdays and weekends and an additional stockpile of batteries (B_{min}) could be maintained on the BSS in case of emergency demand spike. Compared to the statistical model used in chapter 8. This method is more adequate because ARIMA model takes the dynamic of the time series data into consideration

CHAPTER 9 OPTIMIZATION

9.1 INTRODUCTION

When the battery swapping station is operational, it is required to earn maximum profit as possible from the energy stored in the batteries, to do that the BSS needs to supply batteries to meet the demand, and sell back energy to the grid at times where the grid energy demand is higher.

To do that a cost function was derived, where two variables are present,

$$C_c(k) = -(B_{V(k)} \times C_b) + \beta(k) \left\{ \left(B_{Creq(k)} \times \alpha(k) \right) \times C_{b(k)}^g + \left(B_{Creq(k)} \times (1 - \alpha(k)) C_{b(k)}^s \right) \right\} - (1 - \beta(k)) \left(B_{Creq(k)} \times C_{b(k)}^{gs} \right)$$

Here for the k^{th} hour of the day,

$B_{V(k)}$ = Number of batteries demanded by the vehicles

C_b = selling price of batteries to the vehicles

$B_{Creq(k)}$ = Number of batteries that is required to be charged

$\alpha(k)$ = ratio of the number of batteries charged by the grid: generator

$C_{b(k)}^s$ = cost for charging battery via standby generator

$C_{b(k)}^g$ = Cost for charging a battery via the grid energy

$C_{b(k)}^{gs}$ = Selling price of battery energy to the grid

Here $\beta(k)$ is a sigmoid function defined by

$$\beta(k) = \frac{1}{1 + e^{-MB_{Creq(k)}}} \quad (\text{M}>>)$$

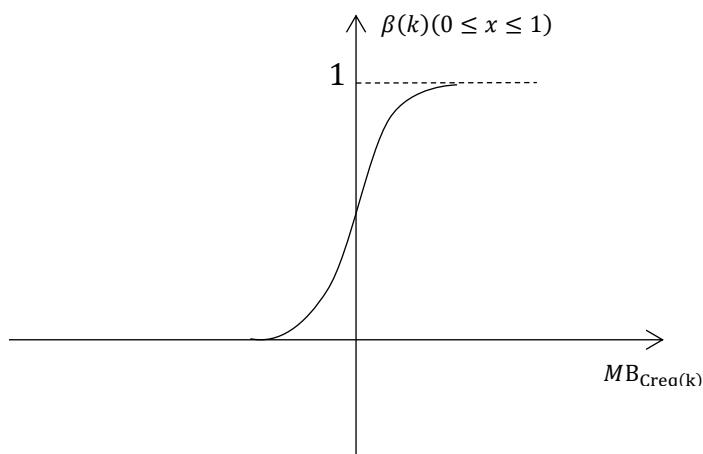


Figure 9.1 Sigmoid function

In the cost function negative value says income while positive value means cost. Target of this optimization is to minimize this cost function as much as possible changing two variables $\alpha(k)$ and $B_{Creq(k)}$

9.2 LAVENBERG MARQURDT ALGORITHM

A MATLAB code is generated to find the minimized cost function by using multiple iterations. The iterative process part which calculates the parameters is shown in the figure 9.2 below.

```
%% INITIAL COST CALCULATION
[cost,beta1,BCreq] = computeCost(Bv,Cb,Cg,alpa,Cs,Badd,Bpv,Cgs);
costOut(1) = cost;
%% Levenberg-Marquardt algorithm
for j=1:iterations

StepRateB = zeros(2, 2, 1, 24); %4d matrix array
StepRateB = findB( beta1,BCreq,Cg,Cs,alpa,Cgs);

gradBadd = findGradBadd( beta1,BCreq,Cg,Cs,alpa,Cgs);
gradAlpa = findGradAlpa( beta1,BCreq,Cg,Cs,Cgs);

    for k= 1:24
        % the new parameters finding
        temp1(:,:,k) = [alpa(k) ; Badd(k)]-StepRateB(:,:,:,:,k)*[gradBadd(k) ; gradAlpa(k)];
        alpa_T(k)= temp1(1,k);
        Badd_T(k)= temp1(2,k);
    end

    %Check for constraints
    for i =1:24
        if ((alpa_T(i) >= 0) && (alpa_T(i) <=1))
            alpa(i) = alpa_T(i) ;
            %constraint violation upper limit / lower limit address here
        else if ((alpa_T(i) < 0))
            alpa(i)=0;
        else if ((alpa_T(i) >1))
            alpa(i)= 1;
        end
        end
    end
        alpa(i)=max(0,min(alpa_T(i),1));
    end

    %Limit of Badd
    for i = 1:23
        Badd(i)=max(0,min(Badd_T(i),B_T - Bv(i+1)));
    end
i=24;
    Badd(i)=max(0,min(Badd_T(i),B_T - Bv(1)));

    %Using new parameters to calculate cost
[cost,beta1,BCreq] = computeCost(Bv,Cb,Cg,alpa,Cs,Badd,Bpv,Cgs);

cost
costOut(j+1) = cost;

end
```

Figure 9.2 Matlab code for the objective function

A data set was generated to test the optimization algorithm as shown in figure 9.3 and the plots are generated for the minimized cost function and the effect of solar generation to this study is also observed by adding a component of solar generation into the chagrining requirement of the batteries.

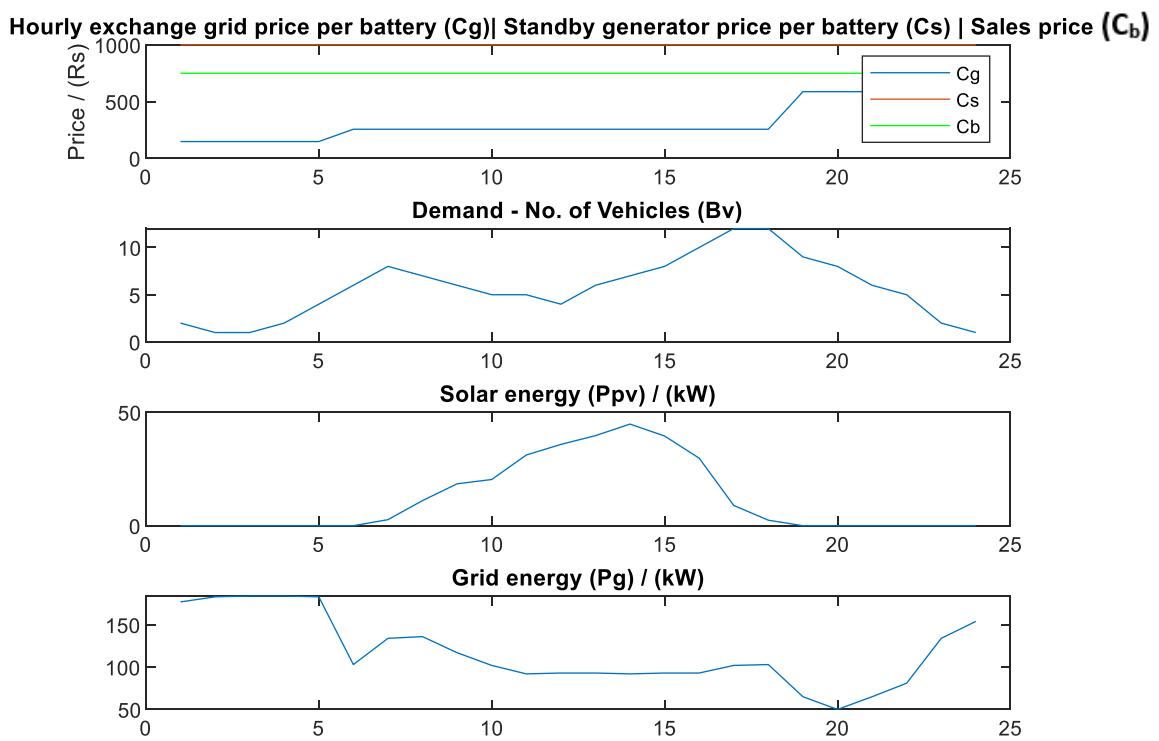


Figure 9.3: (a) Daily variation of price per battery using grid (C_g) & standby generator (C_s) & sales price (C_b)
(b) Daily variation of battery demand demand (B_v)
(c) Solar energy variation throughout the day (P_{pv})
(d) Grid energy available throughout the day

9.3 RESULTS

For the tested data the cost minimization variation with number of iterations is shown in figure 9.4

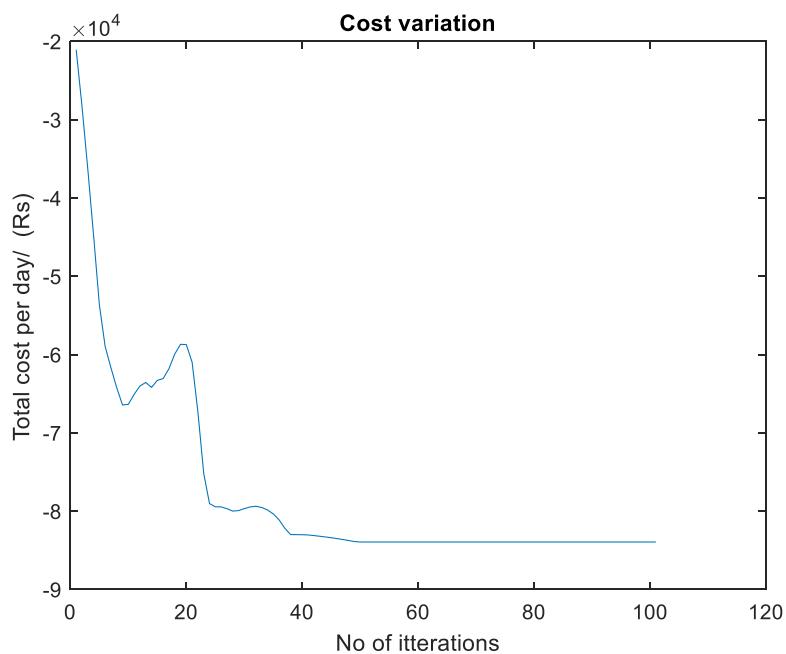


Figure 9.4 Total cost per day variation with the number of iterations

It can be seen that the cost function has reached global minimum value after 40-60 iterations. And the variation of $\alpha(k)$ and the additional battery requirement is shown in the following figure 9.5. Negative Battery requirement implies that the energy can be supplied into the grid

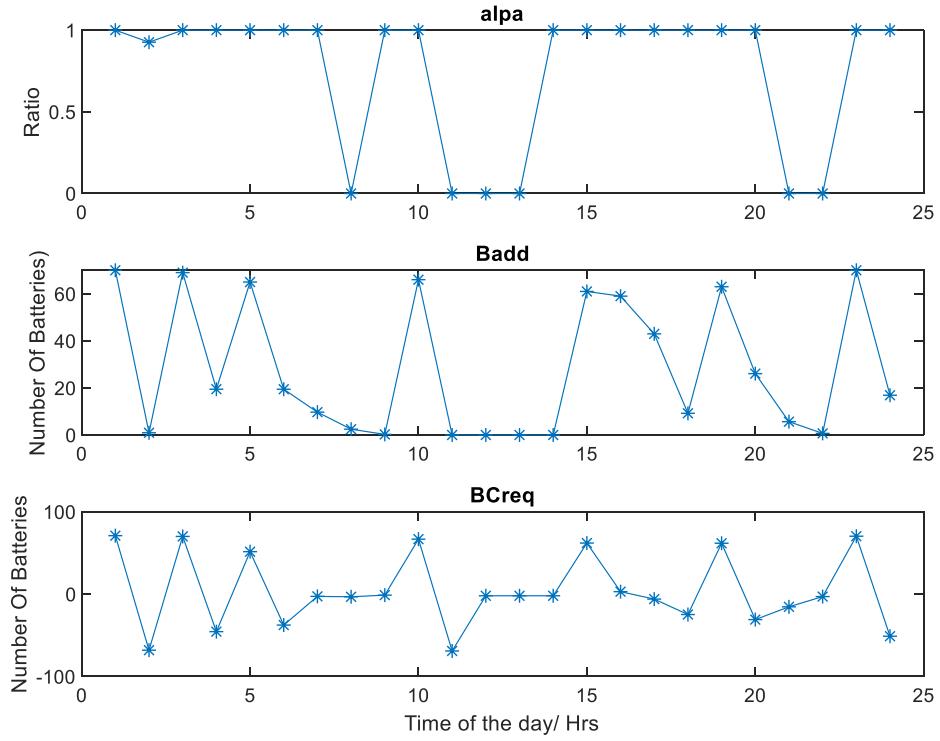


Figure 9.5: (a) variation of $\alpha(k)$ throughout the day

(b) Additional batteries needed to be charged throughout the day
(c) Hourly Additional battery requirement

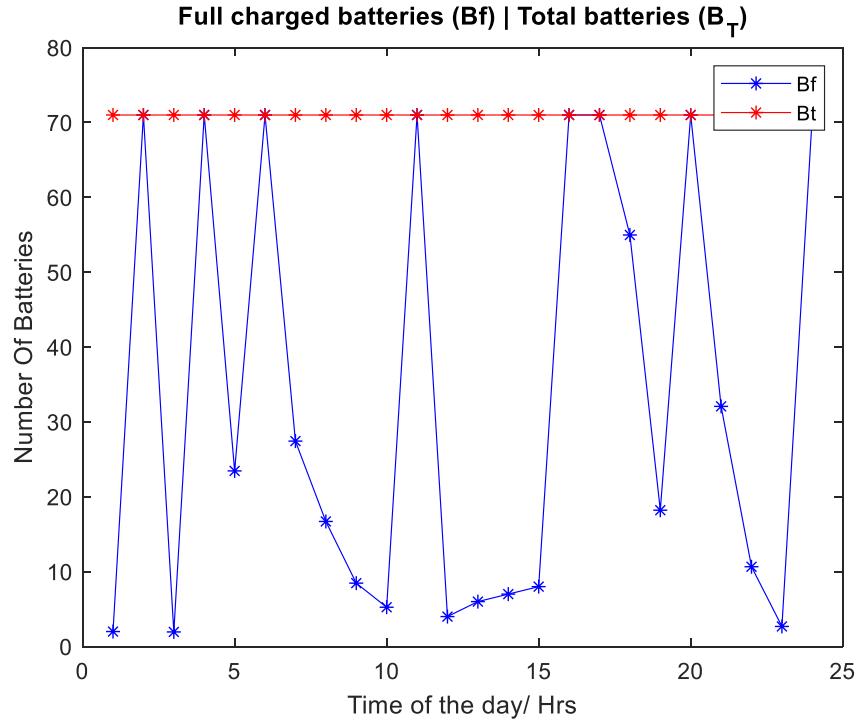


Figure 9.6 Variation of the fully charged batteries throughout the day

And the variation of the energy and the variation of income throughout the day is shown in figure 9.7 below. The negative values in the graphs indicate the energy is supplied back into the grid and, profit is made by selling to the grid.

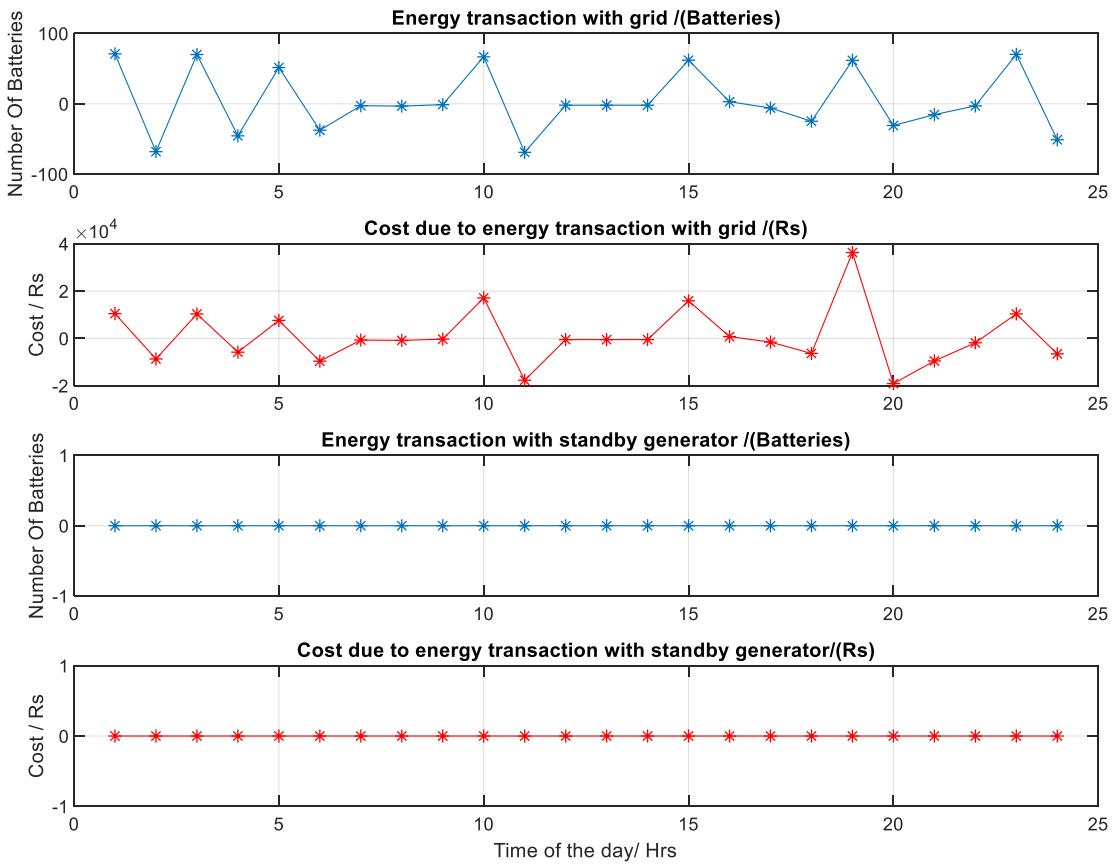


Figure 9.5: (a)Number of batteries used to transfer energy with the grid

(b) Cost or income from the transfer of energy with the grid

(c)Number of batteries used to transfer energy with the standby generator

(d) Cost or income from the transfer energy with the standby generator

9.4 Discussion

By observing the demand curve in figure 9.3, the demand is gradually increasing from 1am to 6pm and since there is no solar energy present, the demand is supplied solely by the grid and the standby generator. Since the grid cost is lower in these hours than the standby generator, only the grid energy is used to charge the batteries. ($\alpha = 1$).

And at 6am to 8am the there is no additional battery requirement, since the energy is sold to the grid and, as seen on figure 9.5 (b) the additional fully charged batteries is gradually dropping. And due to the availability of the solar energy, the additional batteries requirement is charged via the solar energy, that is sufficient to cater demand. And it is also observed that at the peak demand times for the grid which is around 8pm, the algorithm works by selling battery energy into the grid to earn maximum profit.

9.5 CONCLUSION

Upon testing with various grid energy, solar energy, standby generator prices the functionality of the algorithm is tested and it can be further verified after practical implementation of the algorithm. This algorithm is compared with the gradient decent method-based algorithm and, compared to that algorithm this takes much lower iterations to reach the global minimum value. Another advantage of using this algorithm is that it needs only to be run once a day to make the daily plan of that corresponding day.

In order to run this algorithm properly it is important to know the demand beforehand, the methodologies discussed in chapter 7 and chapter 8 address this issue.

CHAPTER 10: REPLACEMENT OF THE BUCK BOOST CONVERTER

10.1 INTRODUCTION

To power the EWheeler, the previous setup used supercapacitors and a converter to get a 48V output voltage and 5KW power. But to remove the supercapacitors from the system, buck boost converters can be used to get the desired output power and output voltage. Specifications of the buck boost converters used in this are as follow.

Table 10.1 : Specifications of the Buck Boost converter

Buck Vdc Input	48 V
Buck Vdc Output	23 V
Buck Idc Output	217 Amp
Boost Vdc Input	23 V
Boost Vdc Output	48 V
Boost Idc Output	105 Amp
Switching Frequency	25 KHz
Fundamental Frequency	50 Hz
Type of cooling	Forced Air Cooled

10.2 CONVERTER SETUP

The optocoupler used for this circuit is HCPL4502 which supports the high frequency required by the converter. The circuit was initially tested on breadboard circuit and the results showed the output of the optocoupler was inverted as shown in figure 10.1.

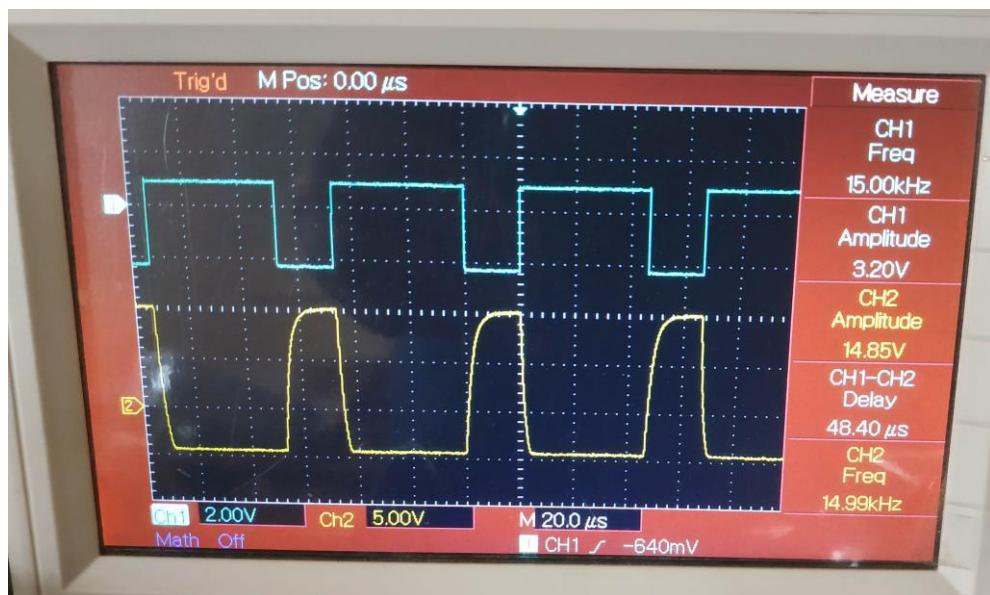


Figure 10.1: Input signal (blue) and the output signal (Yellow)

The circuit optocoupler circuit was designed for the Texas board which outputs 15V, 25kHz Square wave PWM signal, using eagle PCB.

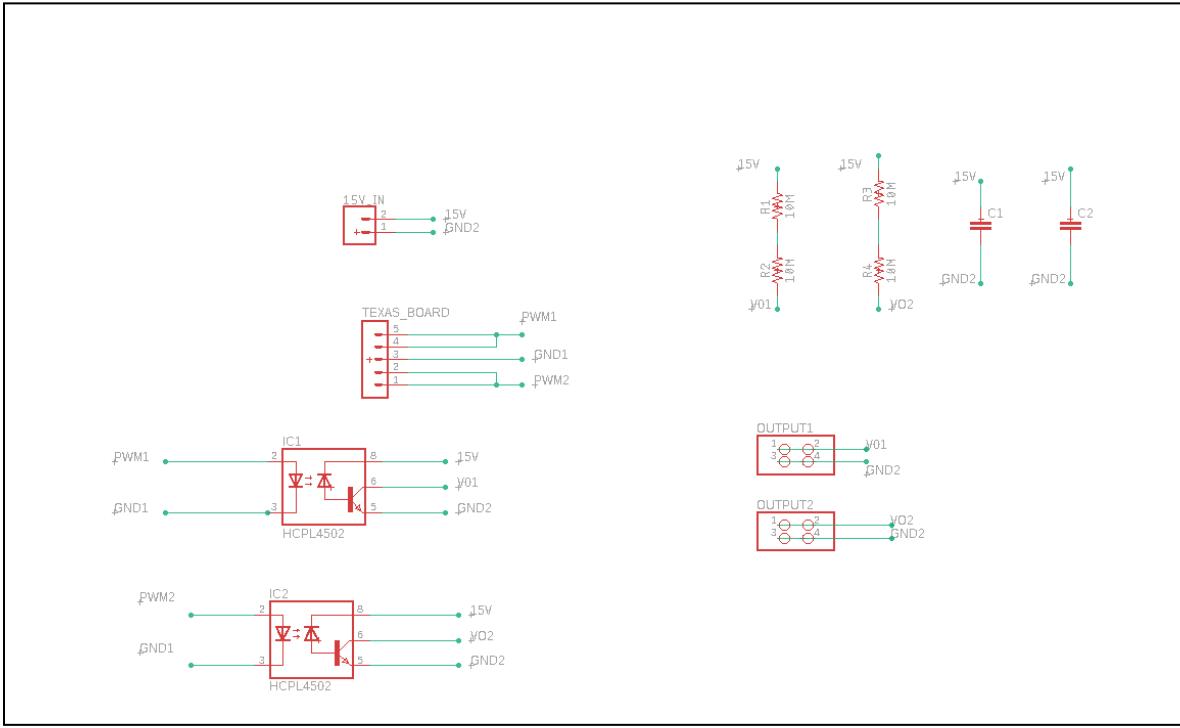


Figure 10.2: Circuit design for the Optocoupler circuit

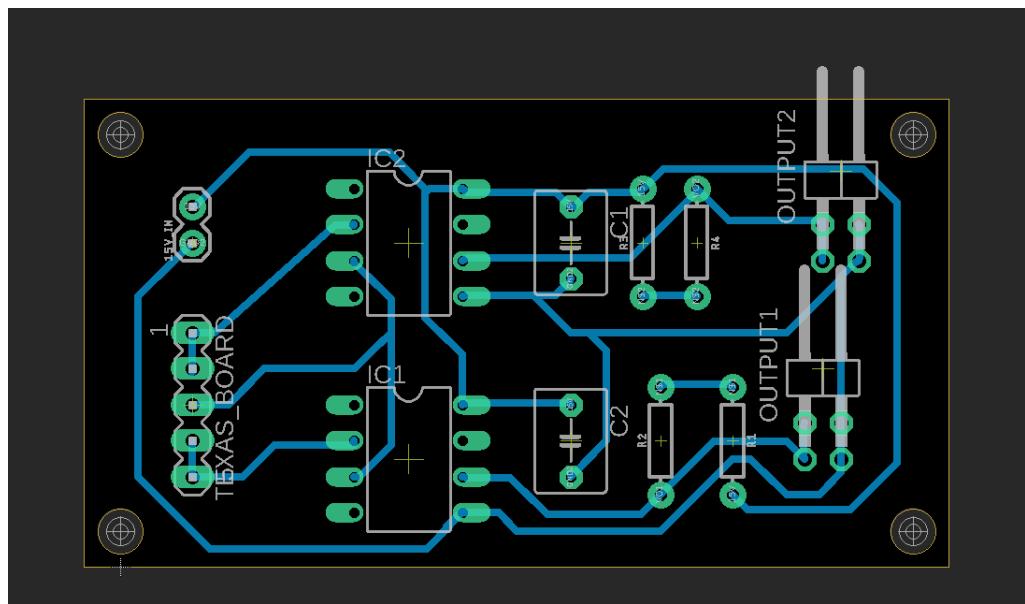


Figure 10.3: PCB Design for the optocoupler circuit

The circuit was tried to make using the CNC available in the DEEE but the drill bit available does not compatible with the circuit designed. Which resulted in very thick trace paths (due to the large size of the CNC Bit).

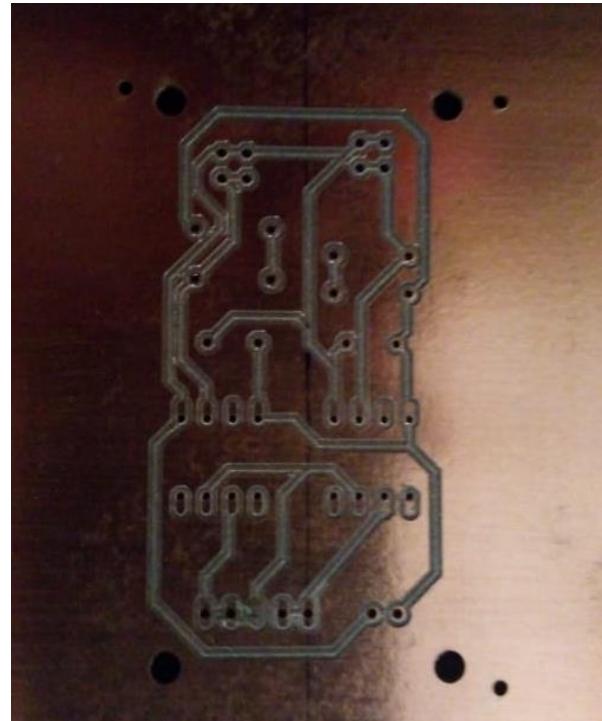


Figure 10.4: Circuit made using CNC

So, as alternative the circuit was tried to create by using UV method. Where the created negative as shown in figure 10.5.



Figure 10.5: Created Negative for the UV method

The UV circuit is yet to be made.

For the initial testing of the converter, the circuit was made using dot board as shown in the figure 10.6.

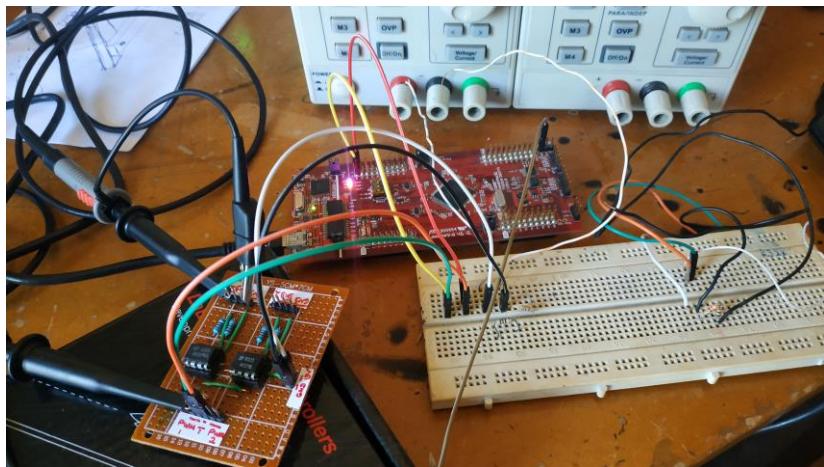


Figure 10.6: Dot Board Circuit

To correct the issue with the inverted output, as shown in figure 10.1, The SOLIDTHINKING EMBED code was modified to invert the PWM signal as shown in figure 10.7.

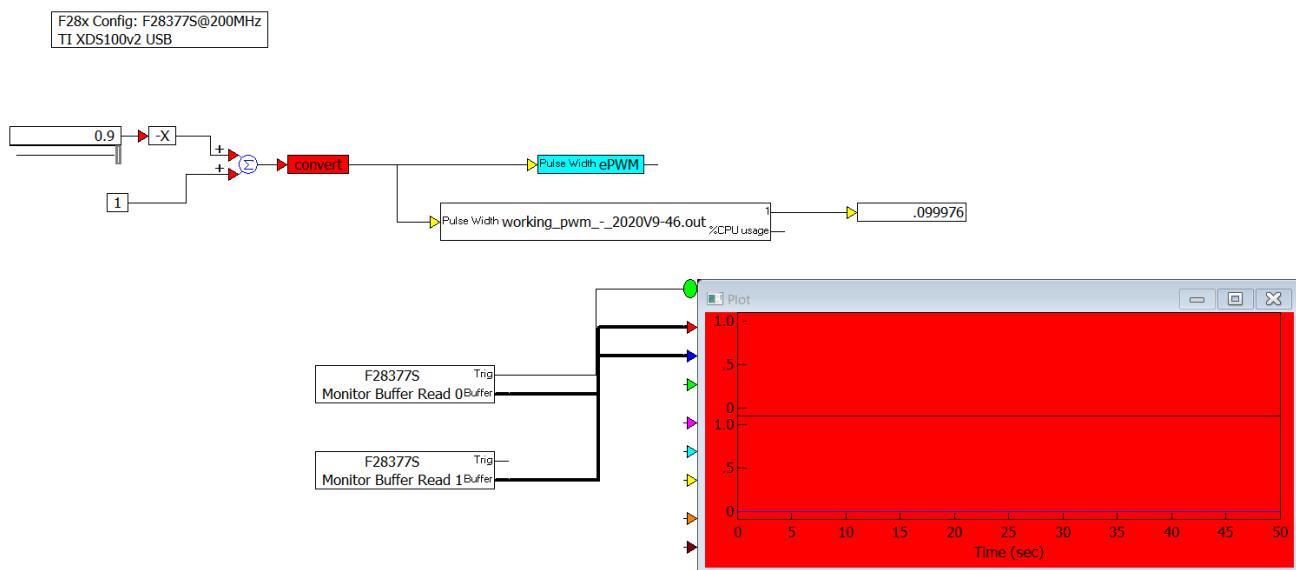


Figure 10.7: SOLIDTHINKING EMBED code for the PWM modification

The circuit was tested as shown in the following figure 10.8, where the inverted PWM of one signal was obtained by the other optocoupler.

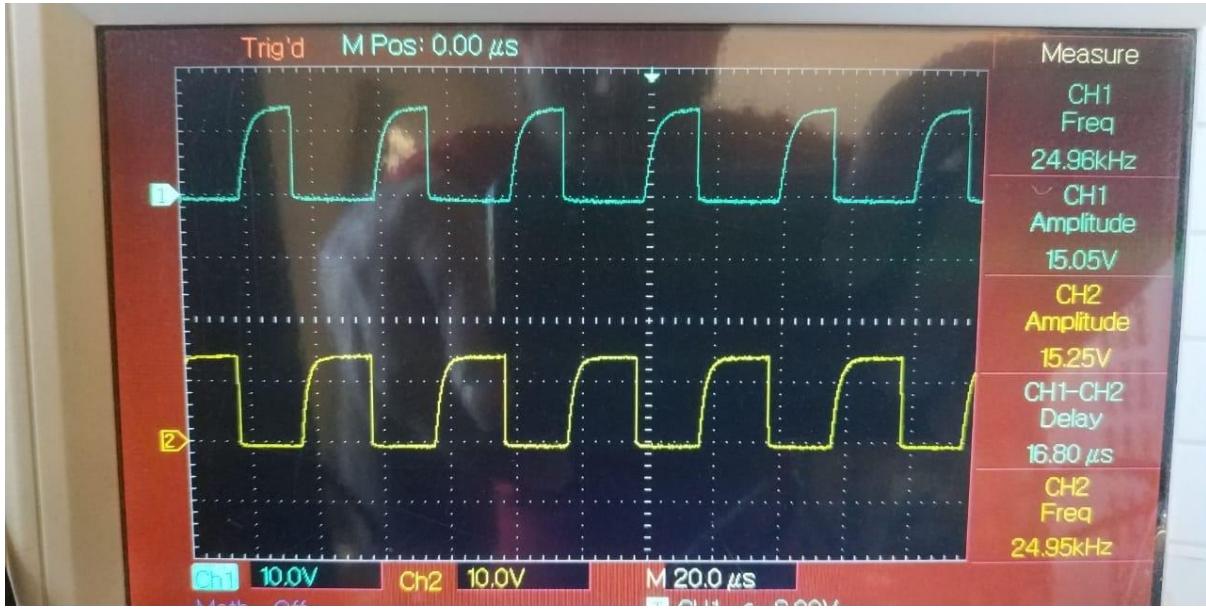


Figure 10.8: Outputs of the two optocouplers

The outputs of the circuit are 15 V and 25 kHz as shown in the above figure.

When observing the skind151041 control board in the Converter, a soldering was observed which seemed unusual as shown in figure 10.9 and figure 10.10.

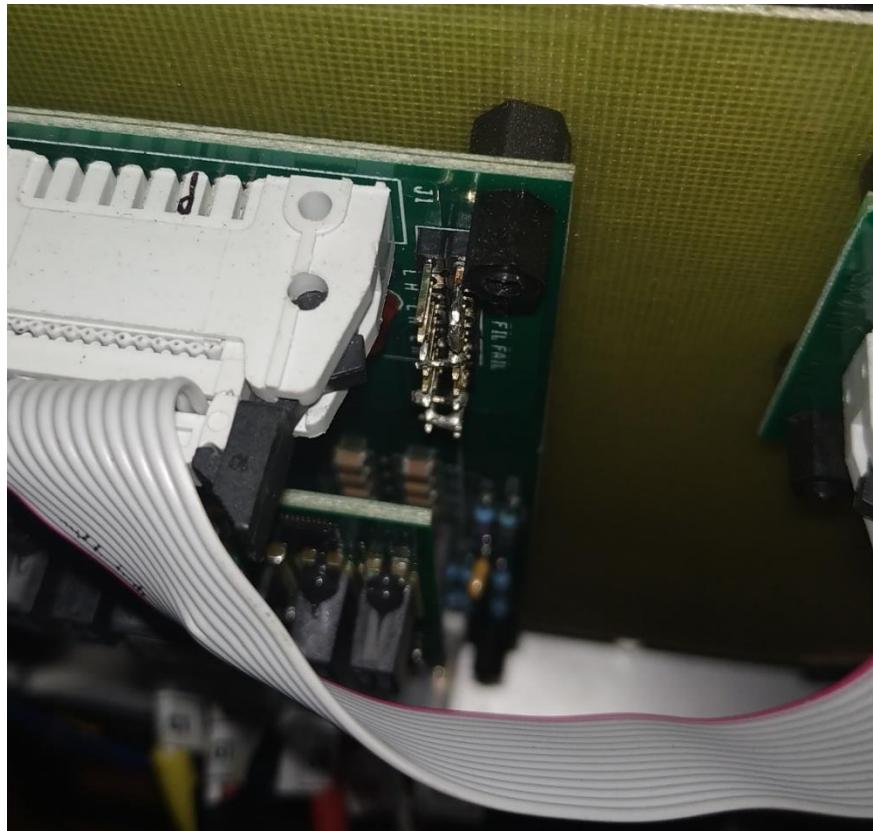


Figure 10.9: observation on the control board I

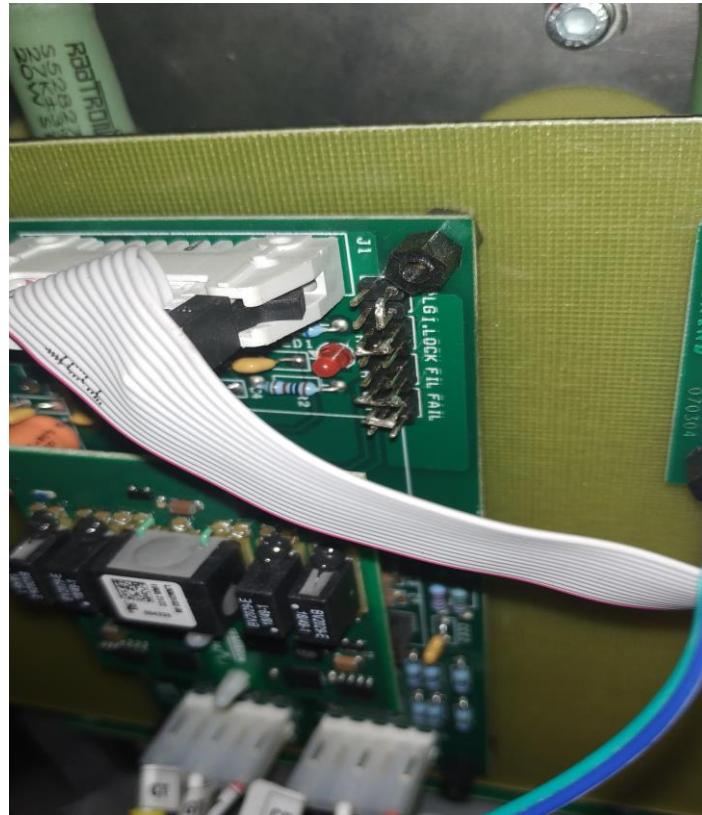


Figure 10.10: observation on the control board II

10.2.1 INTIAL OBSERVATIONS

- At the first trials, it is observed that the IGBT Driver only turns on (RED LED) if there is PWM inputs are connected to the PWM input circuit. The board does not turn on otherwise.
- And as soon after the PWM is applied from the Texas board, the driver turns off (RED LED OFF)
- And the voltage on the PWM pins drops to around 8v (It was 15V and drops soon after it was connected to the controller) as shown in the following figure 10.11.



Figure 10.11: Observations on PWM inputs of the controller board (Vin1 & Vin2)

10.2.2 CURRENT OBSERVATIONS

- The capacitors were connected to the DC supply and the output was observed on the output side. And it was observed that the circuit only seems to work for very short range of PWM (around 0-2%) and does not work otherwise.

So various checks were performed on the controller and, it was finally noticed that the circuit works fine only if,

1. Input to the converter is 15V
2. PWM signal amplitude is 30V (Does not work for 15V as originally thought, the Duty cycle range is limited with lower voltages. To obtain the full duty cycle; the PWM needs to set around 30V)

Input is set to 24V (on the converter) with the above setup.

10.3 TESTING OF THE TWO BUCK BOOST CONVERTERS

The test setup was made with power supply with maximum current rating of 5A and 31V. And a water load setup was used for the testing purposes.



Figure 10.12: Testing setup of the converters

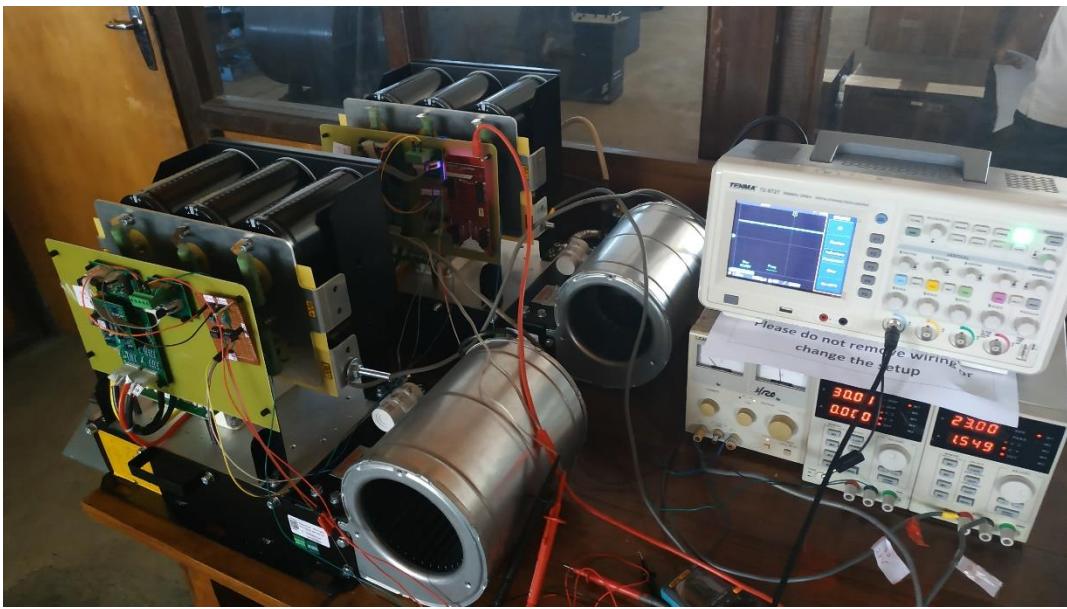


Figure 10.13: Test setup of the converters



Figure 10.14: Water load tested

10.3.1 TEST 1: BOOST MODE OF SETUP 2 WITH 50% \pm 10% DUTY CYCLE WITH DIFFERENT WATER LOAD SETTINGS

Test setting

$V_{in} = 23V$

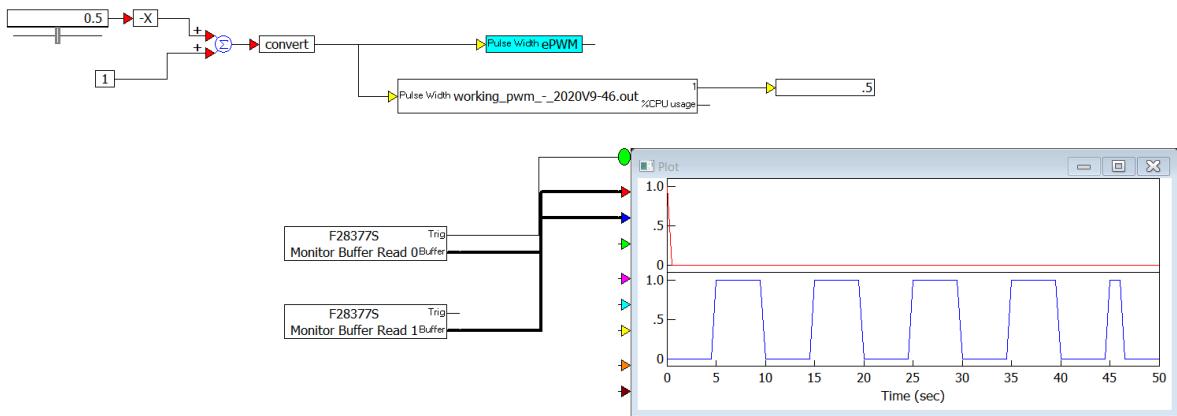


Figure 10.15: Launchpad input with 0.5 duty cycle

Results for 0.5 duty cycle

With the all settings of water load set to low, medium, medium



Figure 10.16: oscilloscope reading



Figure 10.17: current draw reading

With the all settings of water load set to low, low, low



Figure 10.18: oscilloscope reading



Figure 10.19: current draw reading

Results for 0.45 duty cycle

With the power supply limitation, the, load testing was performed with the following settings.

With the all settings of water load set to low, low, low



Figure 10.20: oscilloscope reading

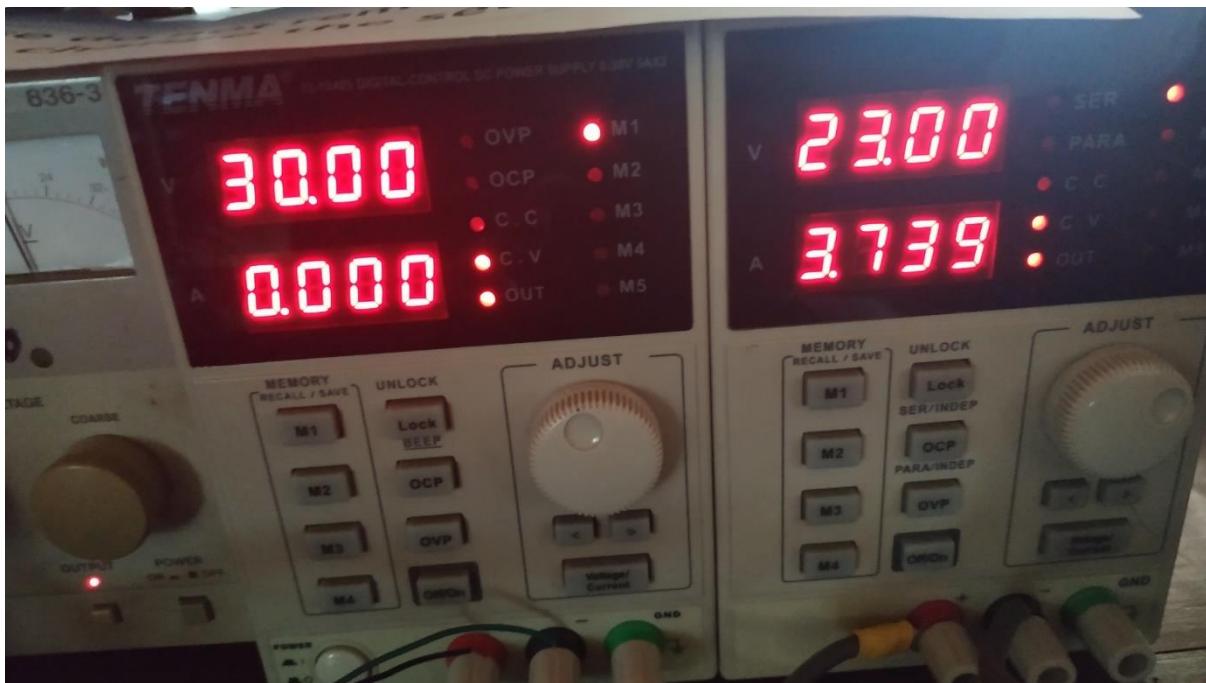


Figure 10.21: current draw reading

With the all settings of water load set to low, low, medium

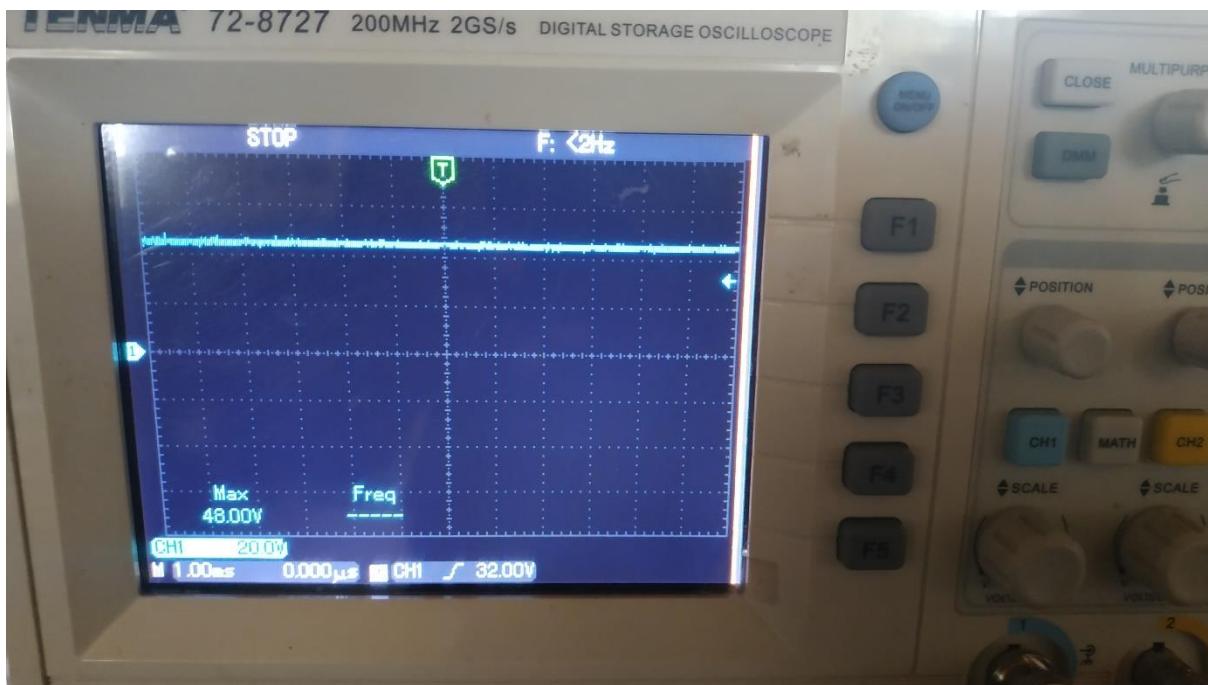


Figure 10.22: oscilloscope reading

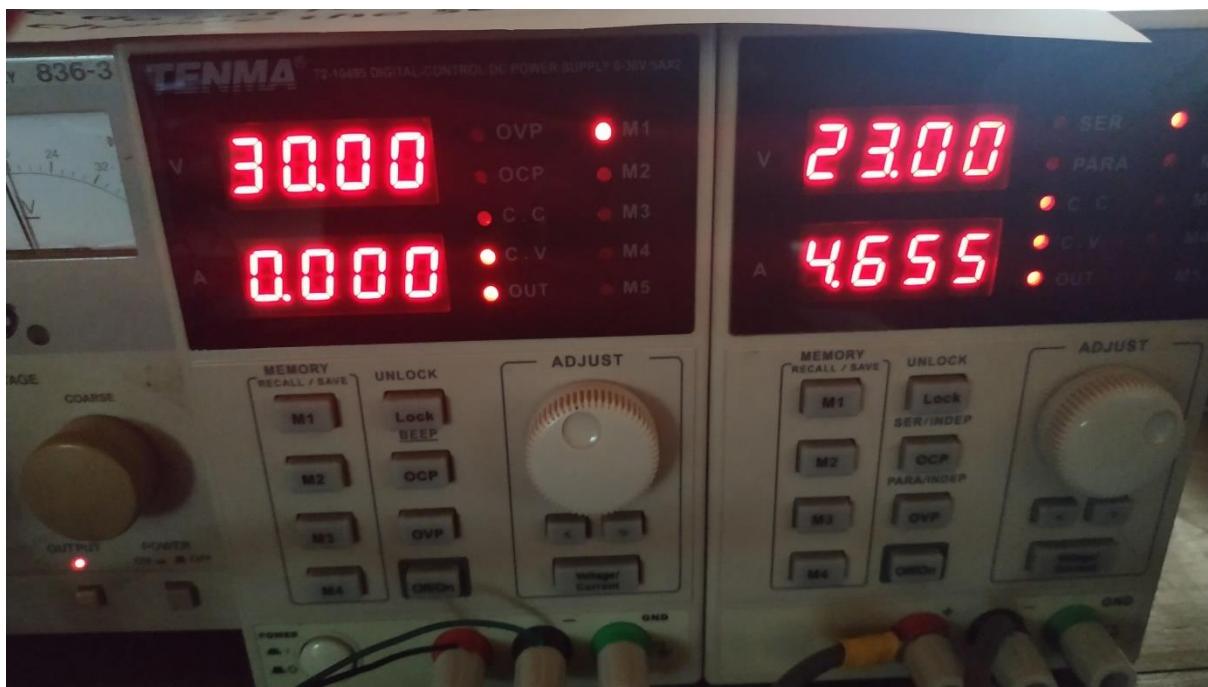


Figure 10.23: current draw reading

Results for 0.55 duty cycle

With the all settings of water load set to low, low, medium

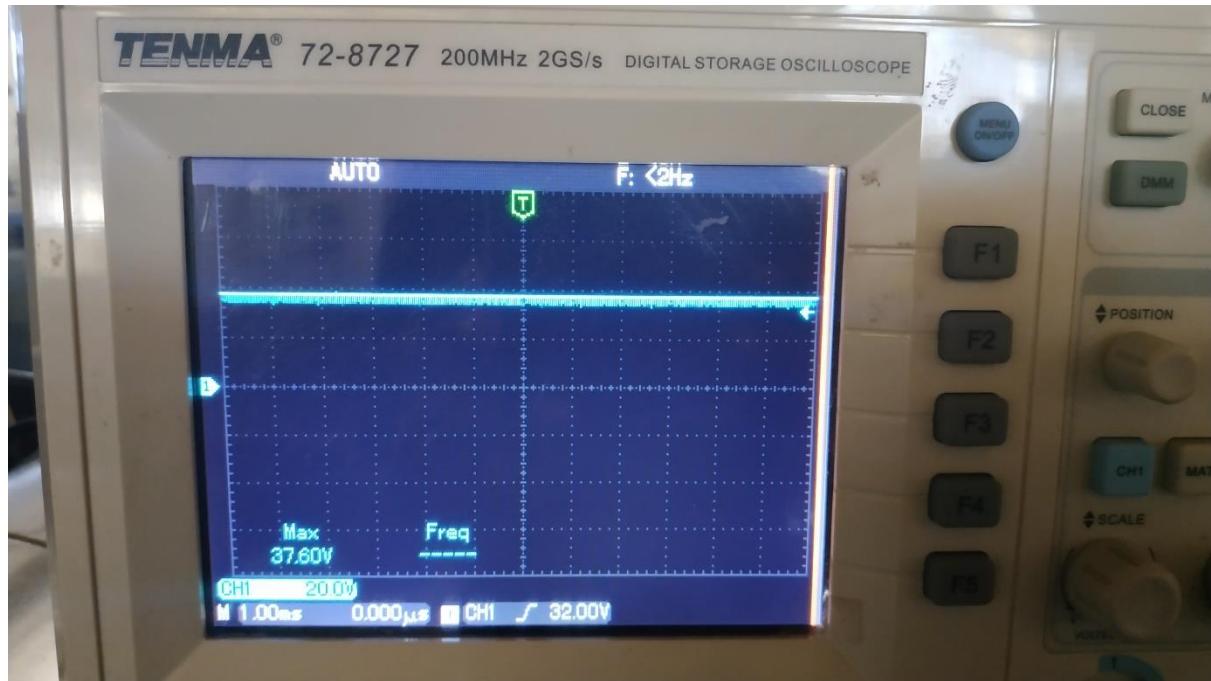


Figure 10.24: oscilloscope reading

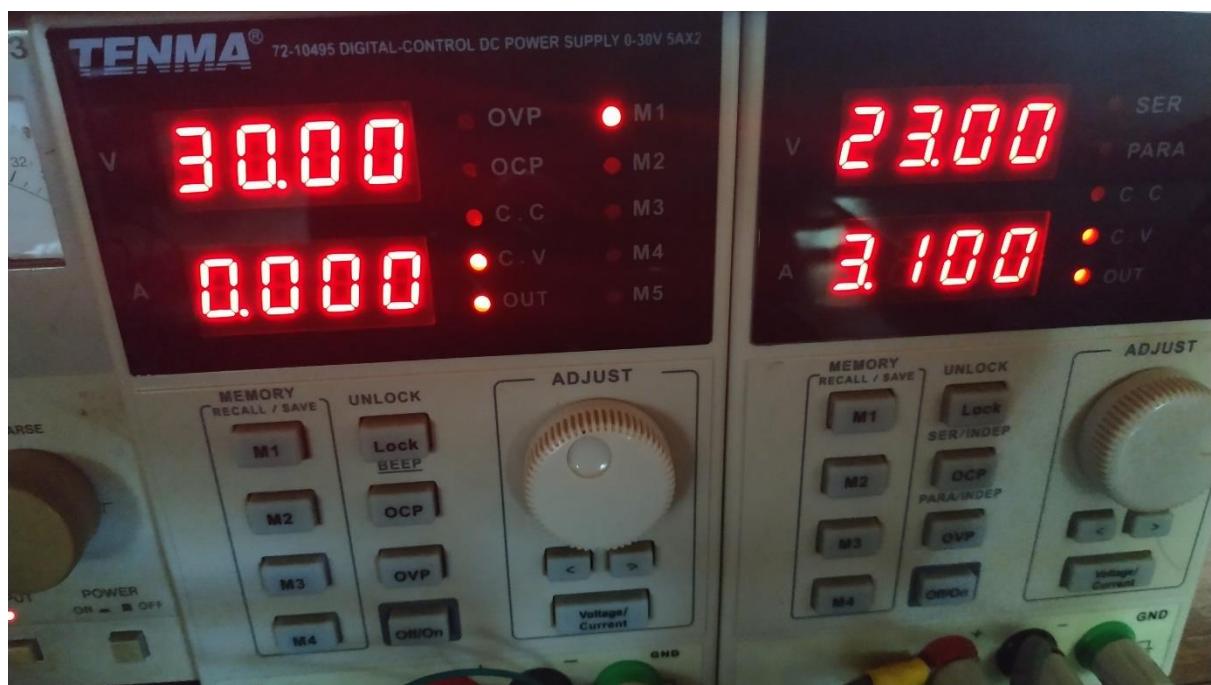


Figure 10.25: current draw reading

With the all settings of water load set to medium, medium, medium

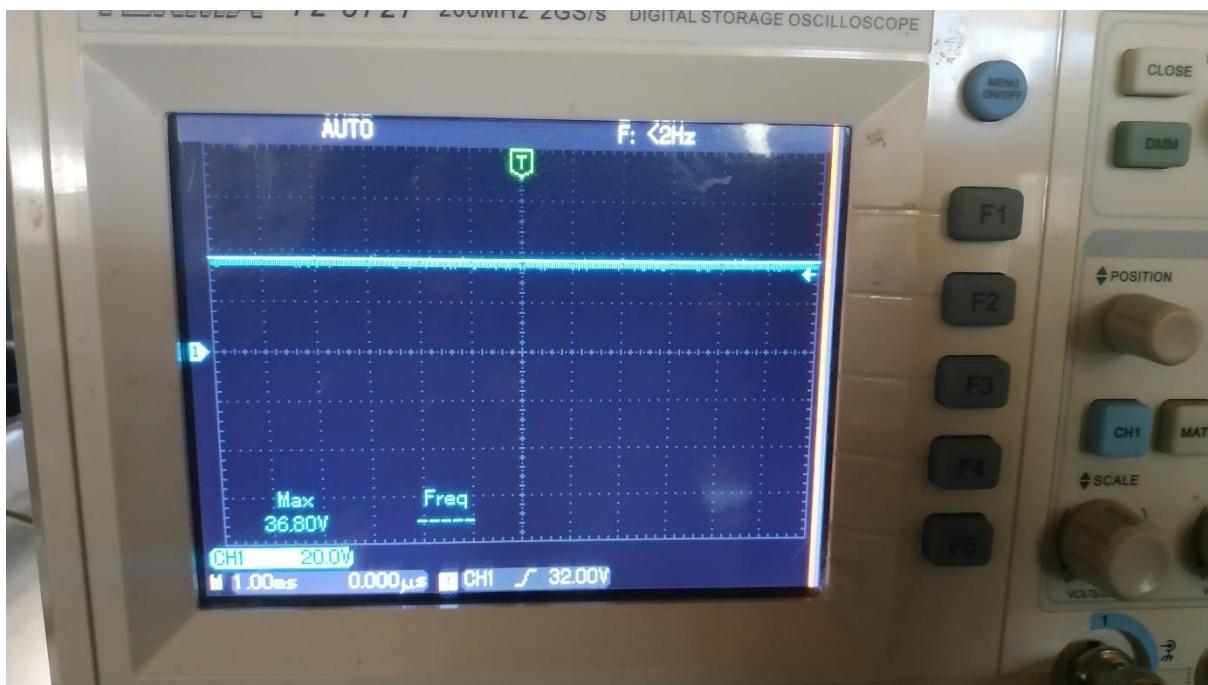


Figure 10.26: oscilloscope reading

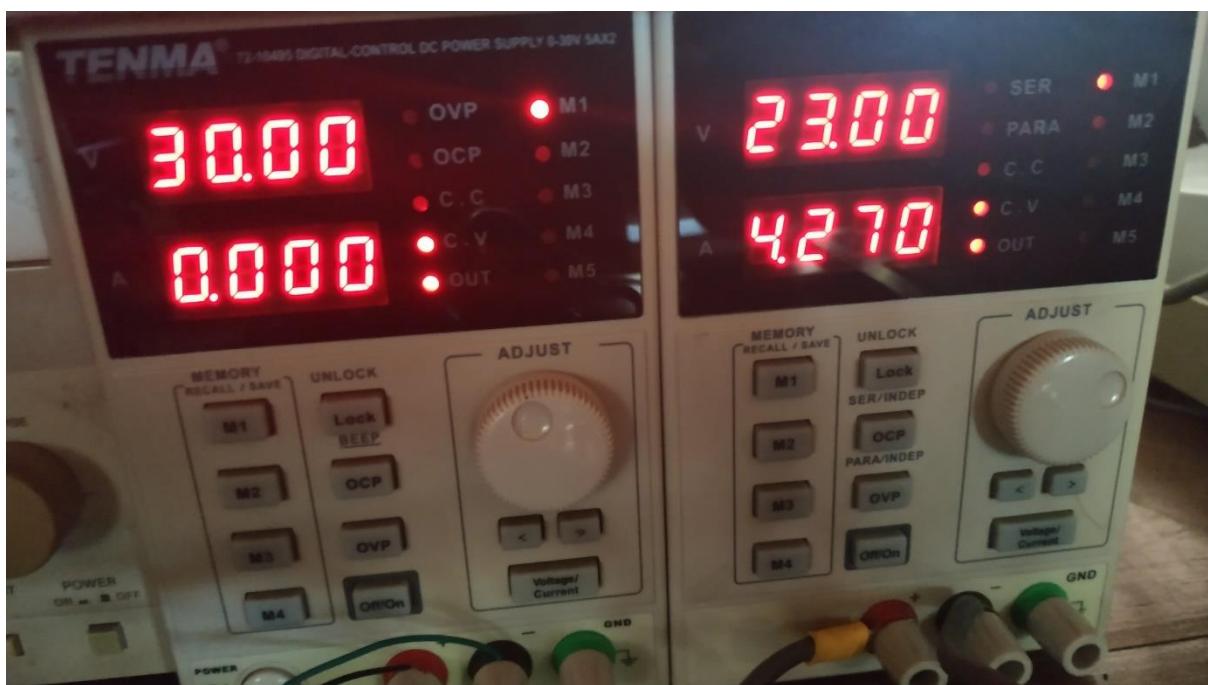


Figure 10.27: current draw reading

10.3.2 TEST 2: BOOST MODE OF SETUP 1 WITH 50% ±10% DUTY CYCLE WITH DIFFERENT WATER LOAD SETTINGS

Test setting

- Vin = 23V

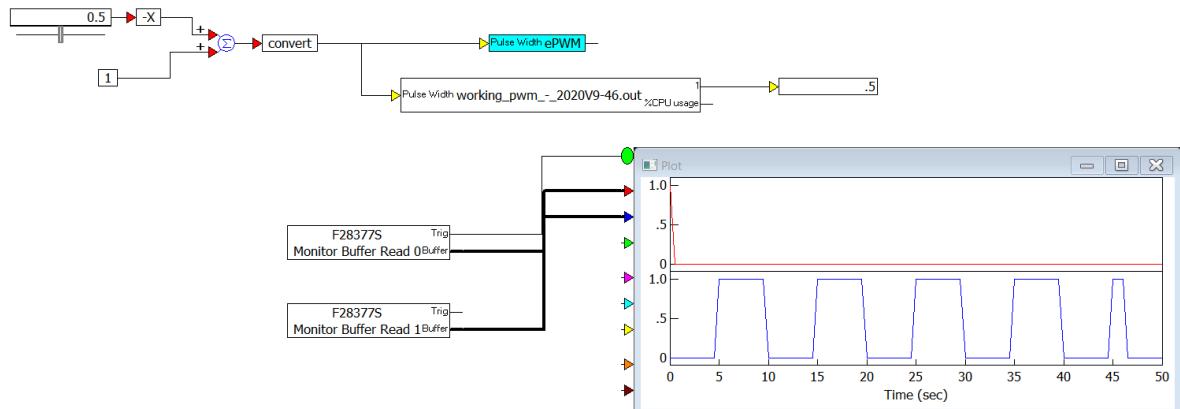


Figure 10.28: Launchpad input with 0.5 duty cycle

Results for 0.5 duty cycle

With the all settings of water load set to low, medium, medium

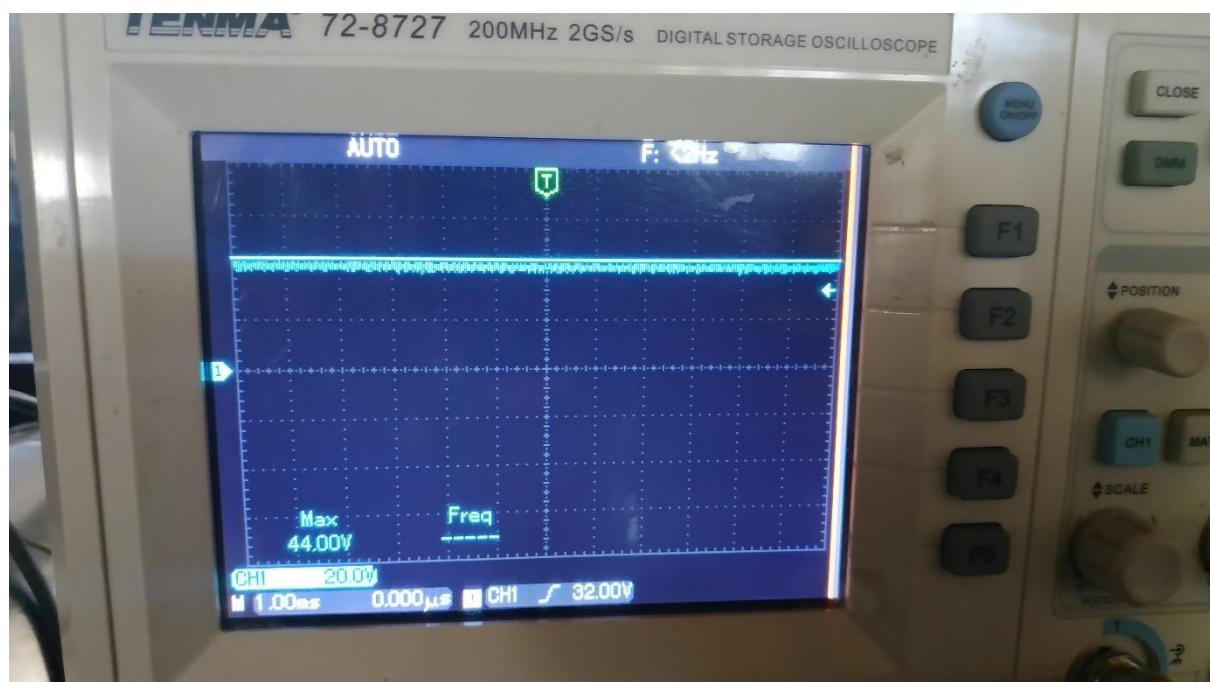


Figure 10.29: oscilloscope reading

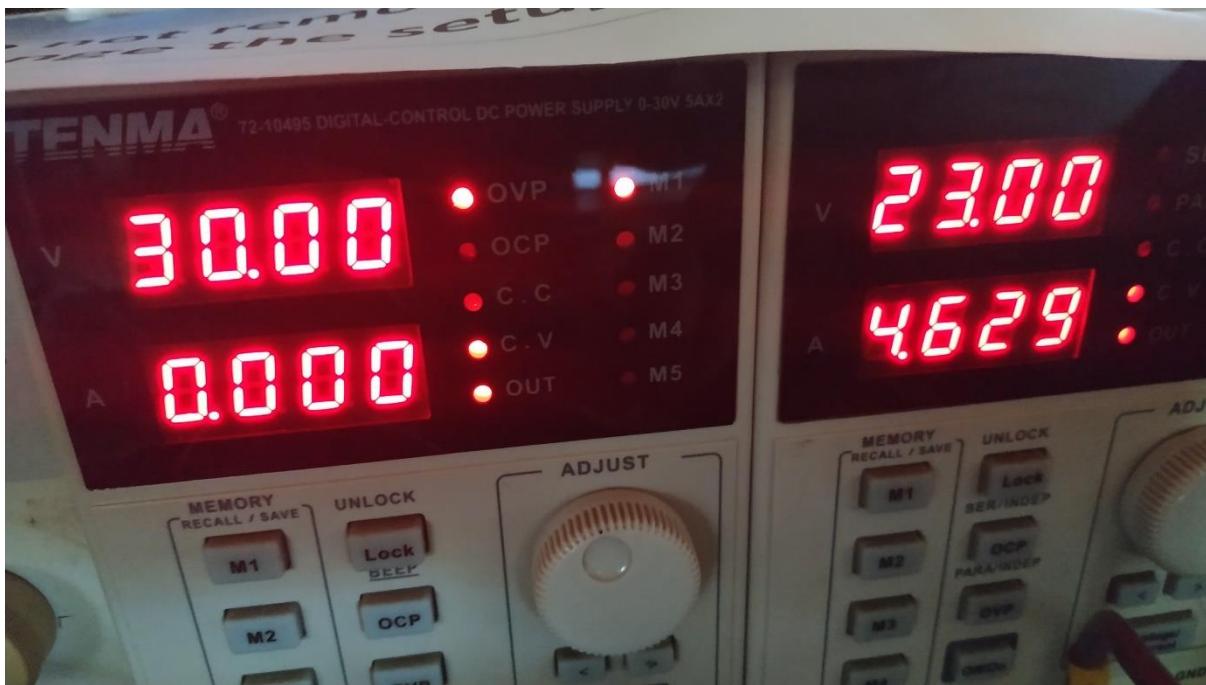


Figure 10.30: current draw reading

With the all settings of water load set to low, low, low

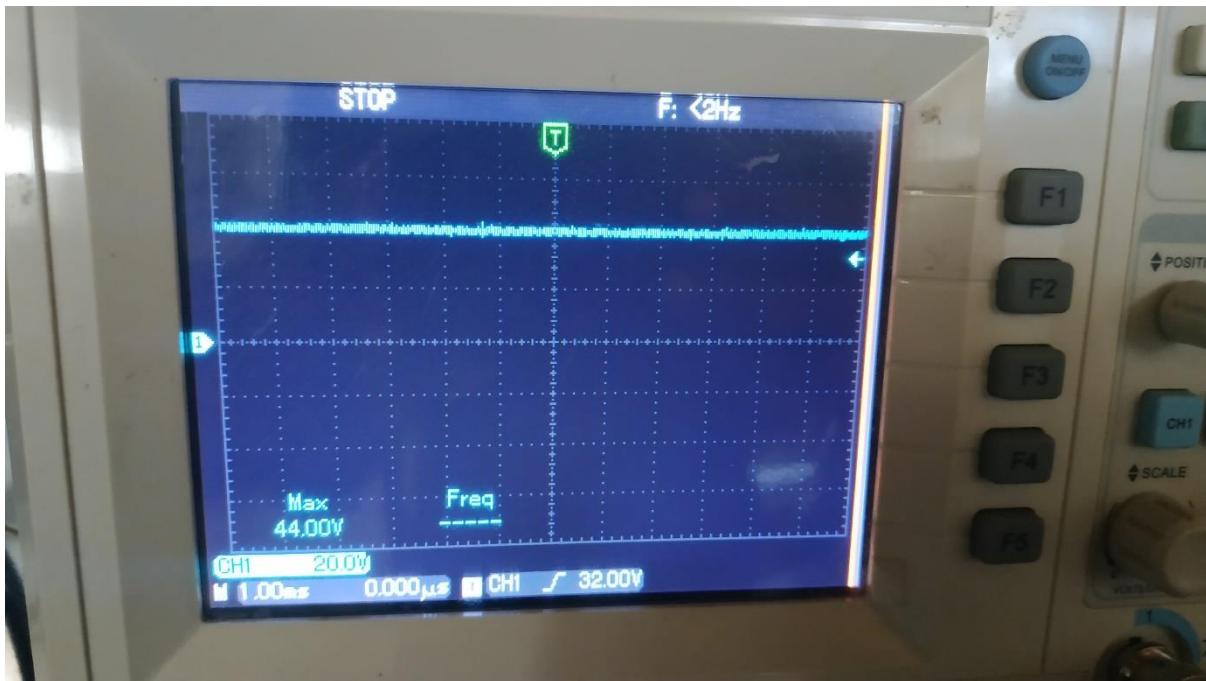


Figure 10.31: oscilloscope reading

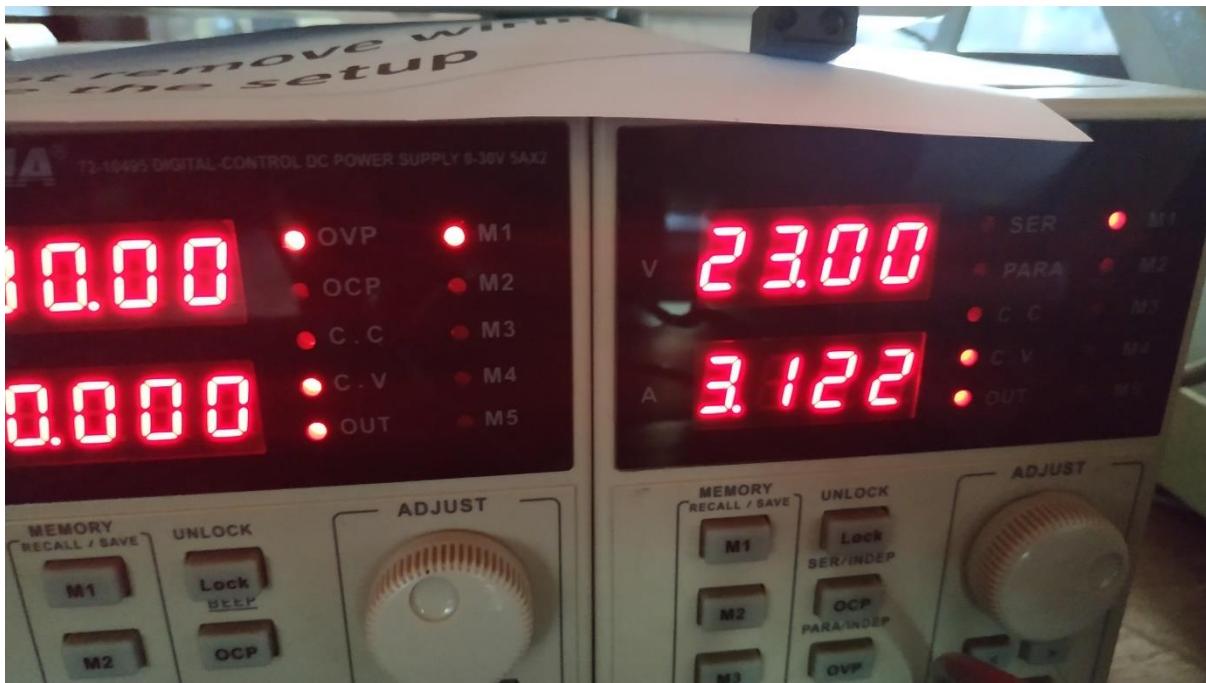


Figure 10.32: current draw reading

Results for 0.45 duty cycle

With the power supply limitation the, load testing was performed with the following settings

With the all settings of water load set to low, low, low

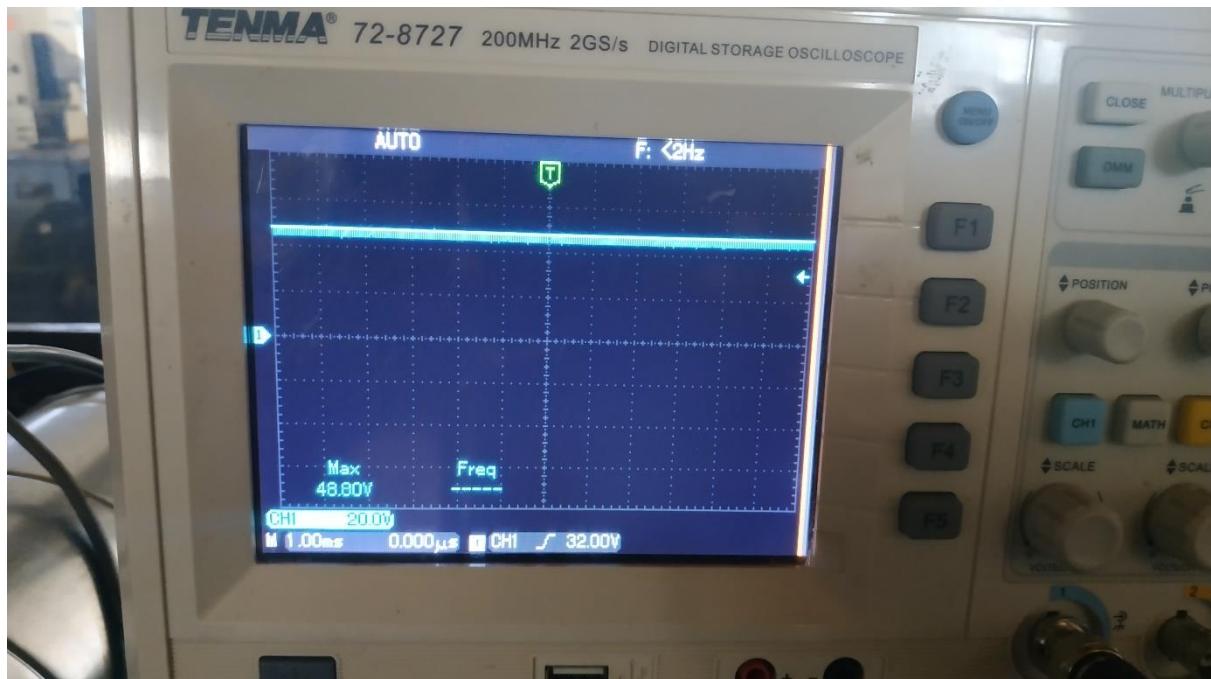


Figure 10.33: oscilloscope reading



Figure10.34: current draw reading

With the all settings of water load set to low, low, medium

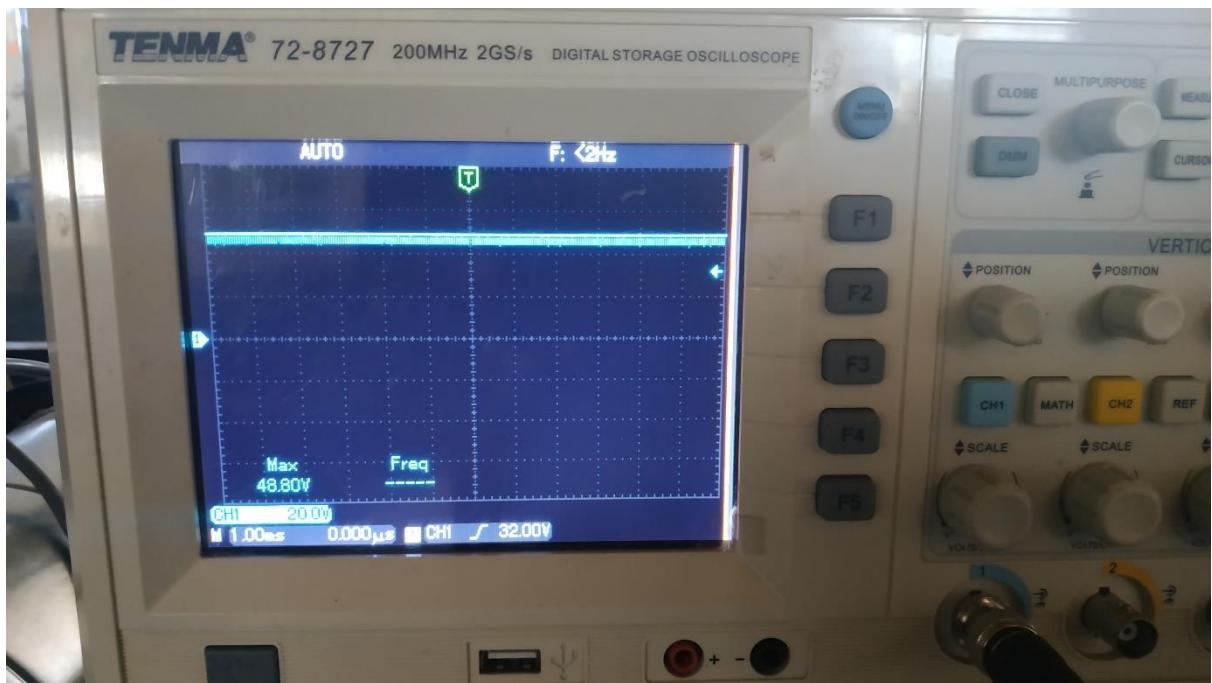


Figure 10.35: oscilloscope reading



Figure 10.36: current draw reading

Results for 0.55 duty cycle

With the all settings of water load set to low, low, medium



Figure 10.37: oscilloscope reading

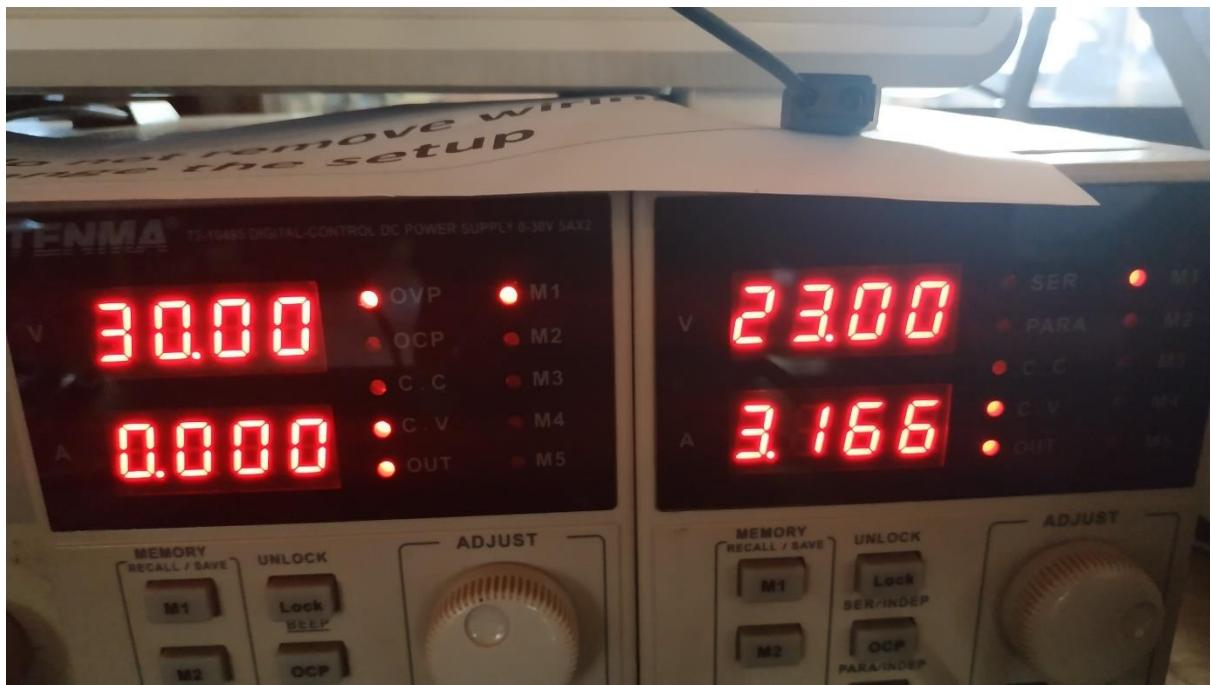


Figure 10.38: current draw reading

With the all settings of water load set to medium, medium, medium

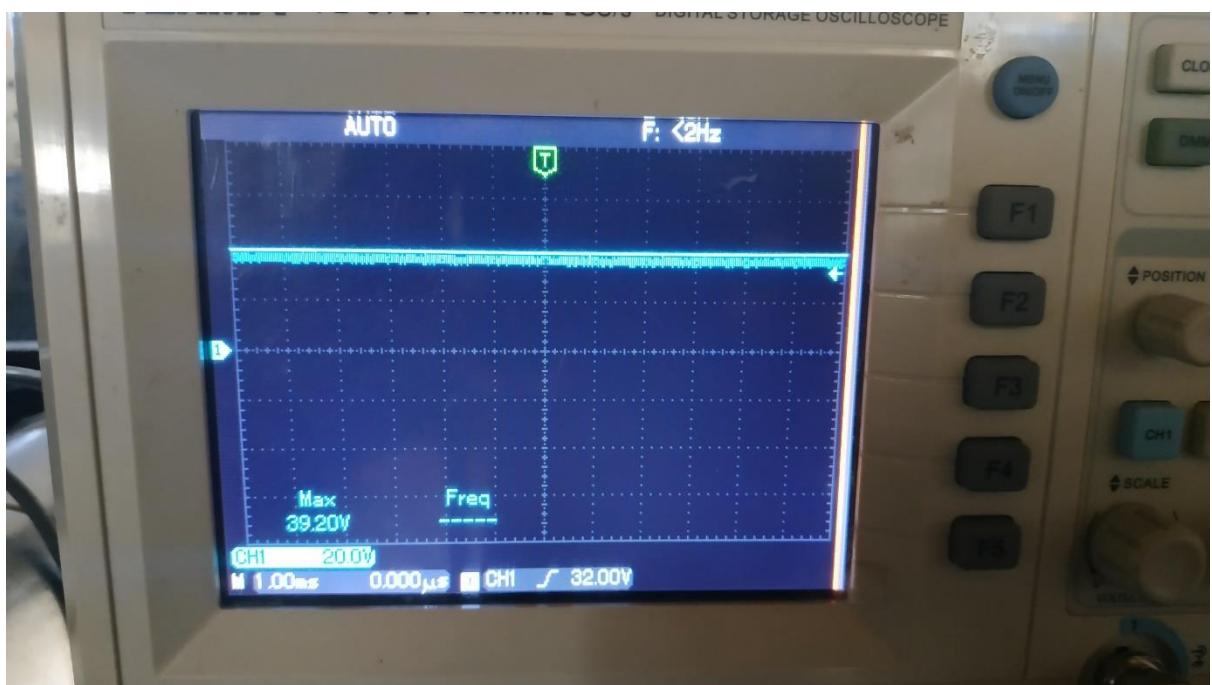


Figure 10.39: oscilloscope reading



Figure 10.40: current draw reading

10.3.3 TEST 3: BUCK MODE TESTING WITHOUT EXTERNAL CAPACITOR FOR SETUP 1

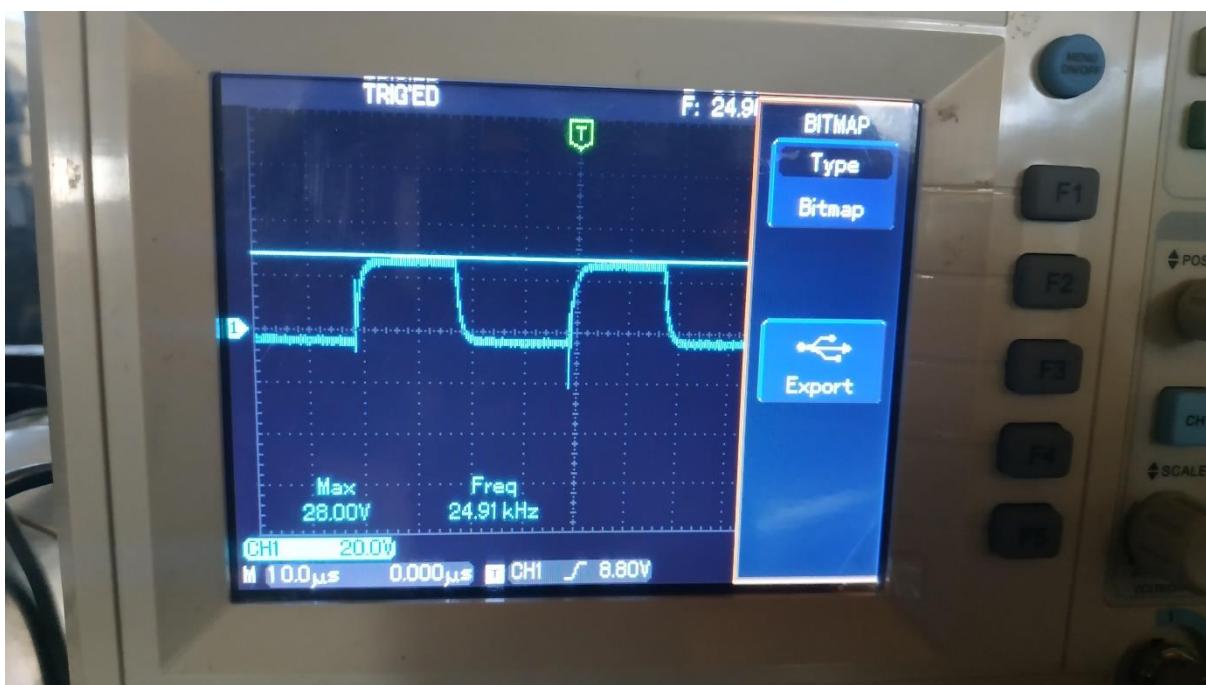


Figure 10.41: Oscilloscope waveform for the buck mode without capacitor

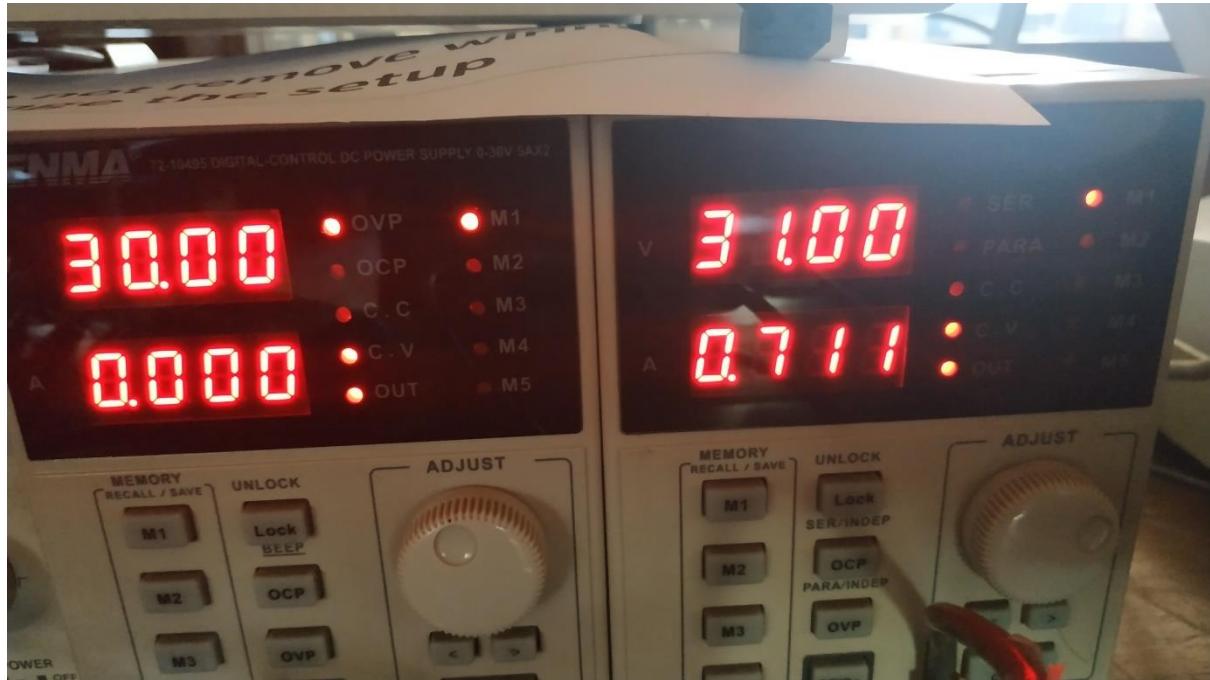


Figure 10.42: Current rating buck mode without capacitor

10.3.4 TEST 4: BUCK MODE TESTING WITH EXTERNAL CAPACITOR (0.5 DUTY RATIO)

Power supply limitation is 31V.

Installed cap 50V, 5000 μ F

Test with 10v input

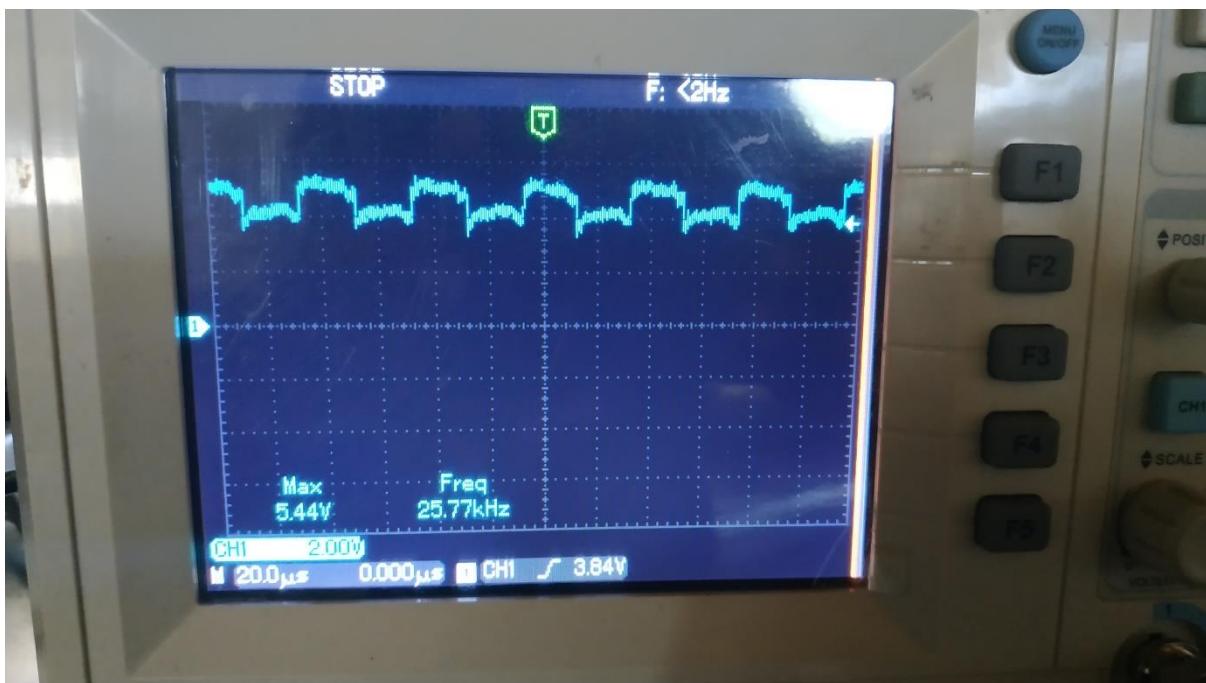


Figure 10.43: Oscilloscope waveform for the buck mode



Figure 10.44: Current rating buck mode

With 18v input



Figure 10.45: Oscilloscope waveform for the buck mode

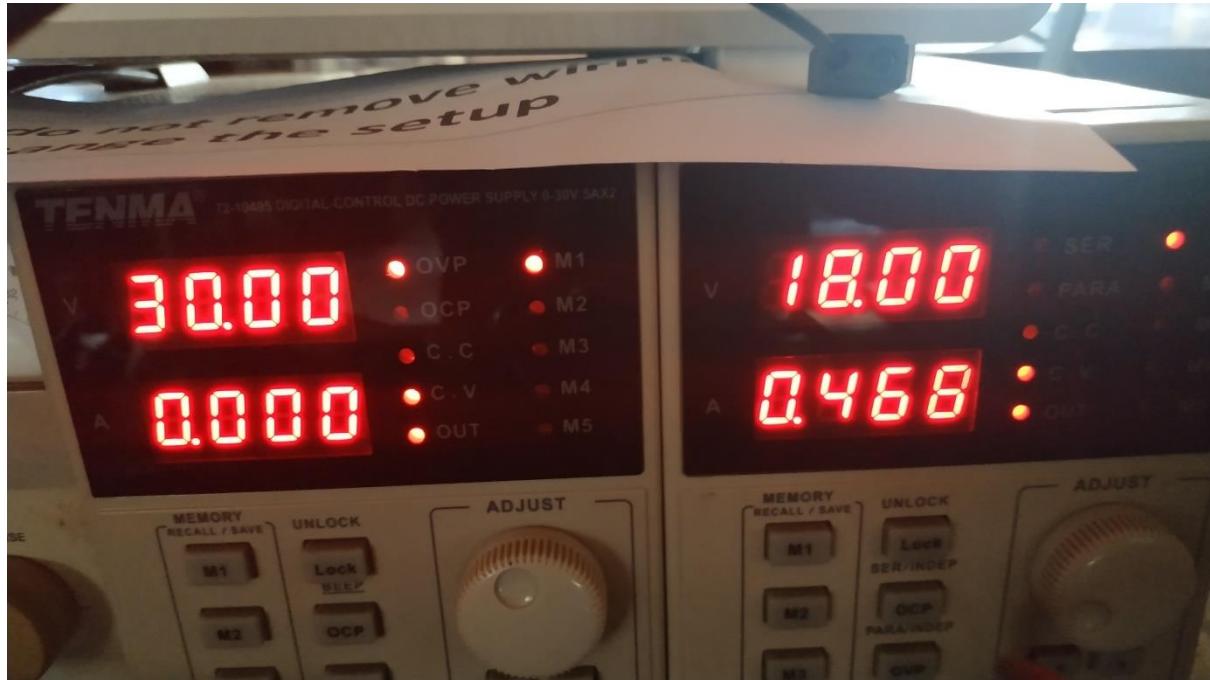


Figure 10.46: Current rating buck mode

With 31v input

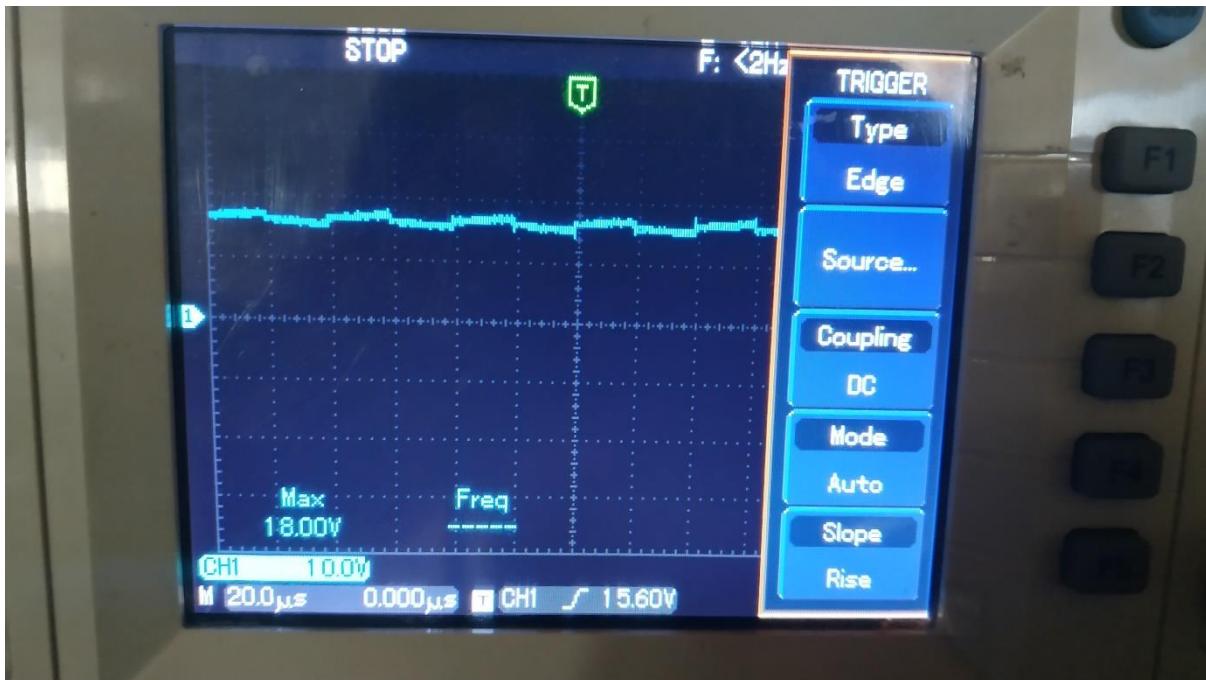


Figure 10.47: Oscilloscope waveform for the buck mode

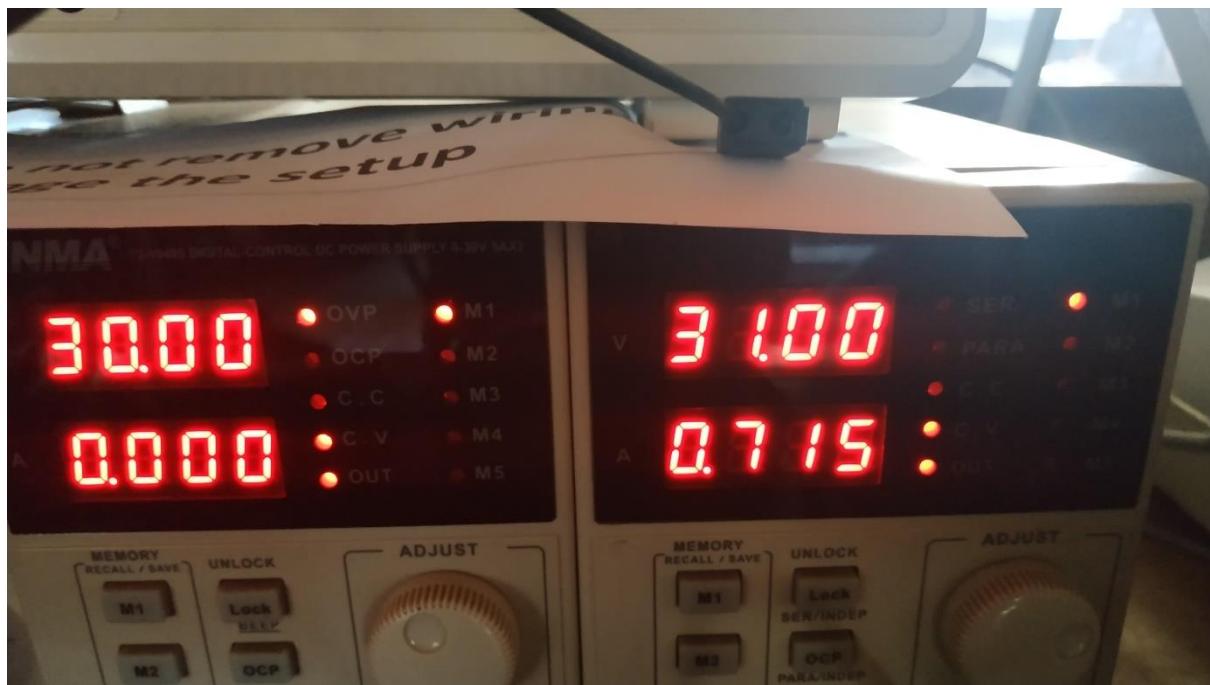


Figure 10.48: Current rating buck mode

10.3.5 TEST 5: BUCK MODE TESTING WITH EXTERNAL CAPACITOR (0.5 DUTY RATIO)

Power supply limitation is 31V.

Installed cap 50V, 5000uF (2x in series)

Test with 10v input



Figure 10.49: Oscilloscope waveform for the buck mode



Figure 10.50: Current rating buck mode

With 18v input



Figure 10.51: Oscilloscope waveform for the buck mode



Figure 10.52: Current rating buck mode

With 31v input



Figure 10.53: Oscilloscope waveform for the buck mode

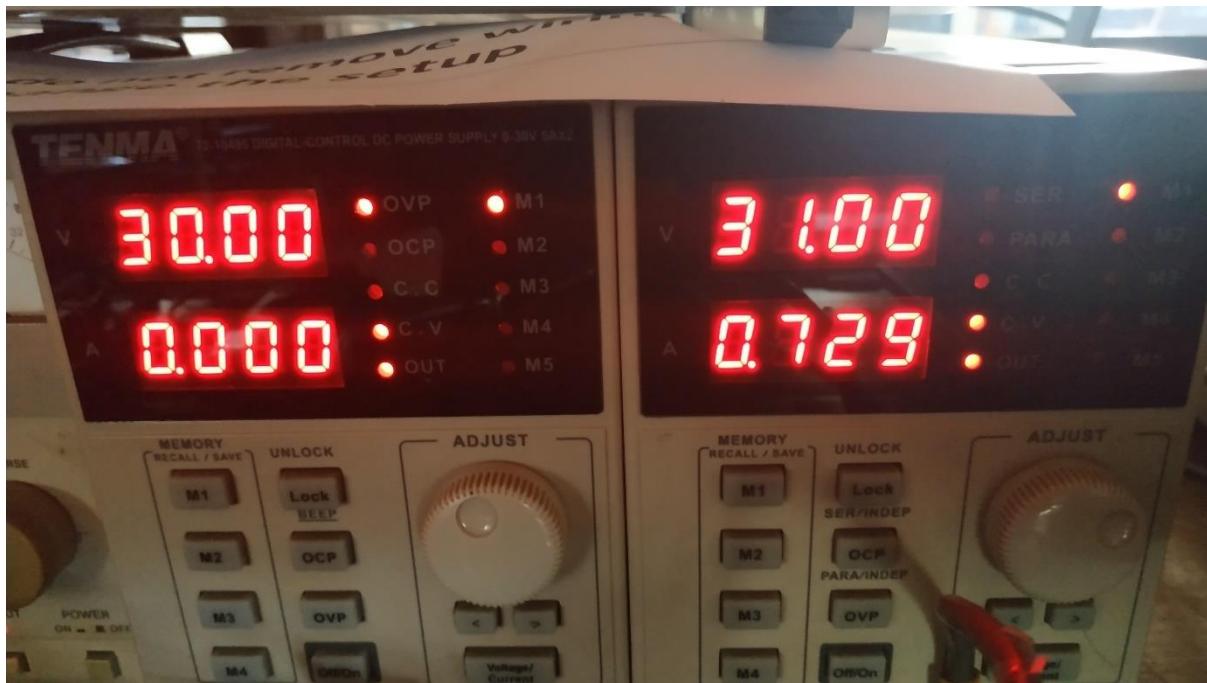


Figure 10.54: Current rating buck mode

10.4 OBSERVATIONS AND RESULTS

The Load Tests were performed with all load settings for the duty ratios of 0.45, 0.5 and 0.55. And the following results were obtained.

Table 10.2: Observations of the Load Testing

Water Load mode	Load resistance Ω	Input voltage V	Input current			Output voltage			Output current			Temperature		
			D=0.45	D=0.50	D=0.55	D=0.45	D=0.50	D=0.55	D=0.45	D=0.50	D=0.55	d=0.45	d=0.5	d=0.55
LLL	42.9	23	304	360	440	42.8	47.2	50.4	110	124	132	29	30	29
LLM	34.6	23	360	440	540	42	46.8	50	144	162	174	29	31	29
LLH	25	23	480	600	700	42	46.4	49.2	216	240	252	30	31	29
LML	33.8	23	360	440	540	42	46.8	50	144	162	172	30	31	29
LMM	27.5	23	424	560	620	42	46.8	50	180	204	214	30	31	29
LMH	20.8	23	552	620	700	41.6	42	46	250	250	278	30	31	29
LHL	27.4	23	488	600	700	42	46.4	48.8	216	238	254	30	31	30
LHM	23	23	576	680	700	42.4	45.6	46	256	274	278	30	31	30
LHH	17	23	656	700	840	41.2	42.4	45.2	316	324	348	30	31	30
MLL	33	23	360	440	520	42	46.4	50.4	142	158	170	30	31	30
MLM	27.3	23	432	520	660	42.4	46.4	50	180	198	214	30	31	30
MLH	20.5	23	560	660	720	42	45.6	46.4	252	276	278	30	31	30
MML	27	23	432	520	620	42.4	46.4	50	182	196	212	30	31	30
MMM	22.8	23	504	600	720	43.2	46.8	48.8	214	238	250	30	31	30
MMH	18.2	23	624	680	780	42	44	46	288	298	314	30	31	30
MHL	21.1	23	560	680	720	42.8	46.4	46.4	260	278	278	30	31	30
MHM	18.4	23	656	720	780	43.6	44	45.6	286	302	314	30	31	30
MHH	15.3	23	600	740	820	36	41.2	42.4	310	350	362	30	31	30
HLL	23.3	23	460	580	700	38.8	46	48.8	200	236	250	30	31	30
HLM	20.5	23	480	660	720	37.6	45.6	46.4	224	272	278	30	31	30
HLH	16.6	23	580	700	820	36.8	42.4	44	284	326	340	30	31	30
HML	20.1	23	480	660	700	37.6	45.6	46	224	272	276	30	31	30
HMM	18.5	23	520	660	780	36.8	43.2	45.6	254	296	314	30	31	30
HMH	15.5	23	600	760	820	36	41.2	42.4	310	352	364	30	31	30
HHL	17.4	23	560	700	820	36.8	42.4	44	284	326	340	30	31	30
HHM	15.7	23	600	760	820	36	41.2	42.4	310	352	364	30	31	30
HHH	14	23	640	820	900	34	40	40.4	352	410	416	30	31	30

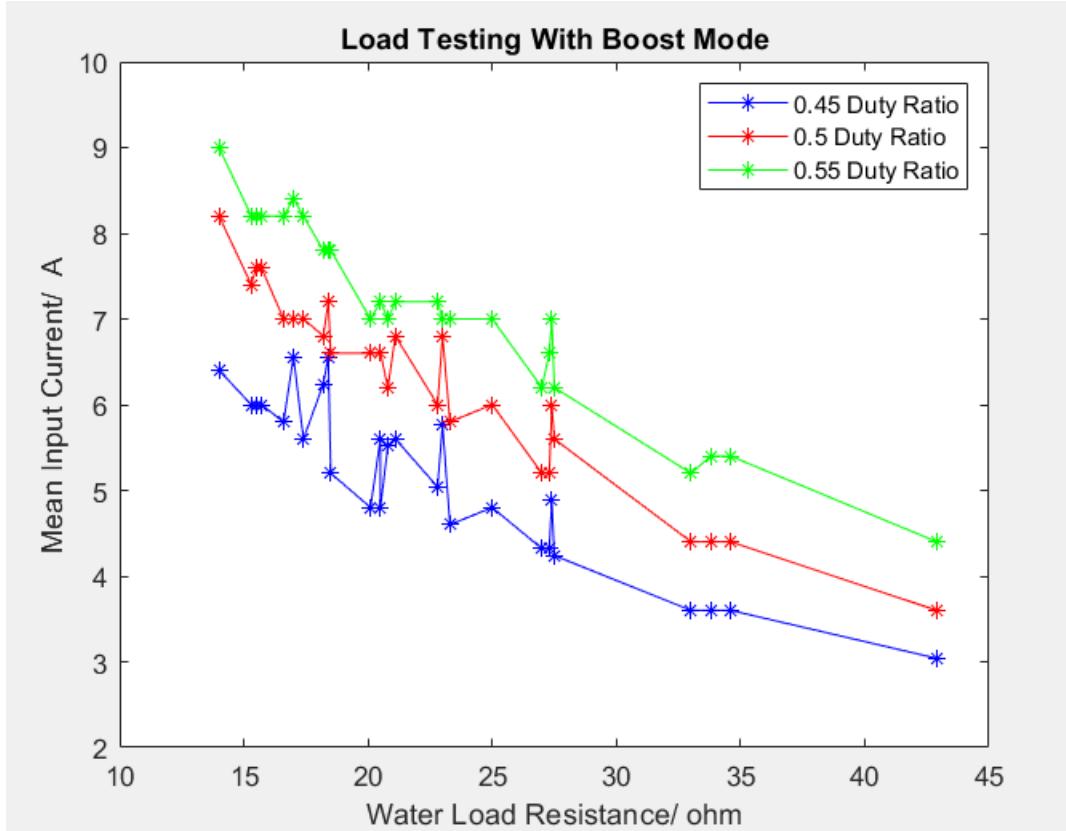


Figure 10.55: Variation of Input current with the Water Load resistance

The results indicated the mean output current decline as the load resistance increases. And with increasing duty ratio. The input current requirement also increases.

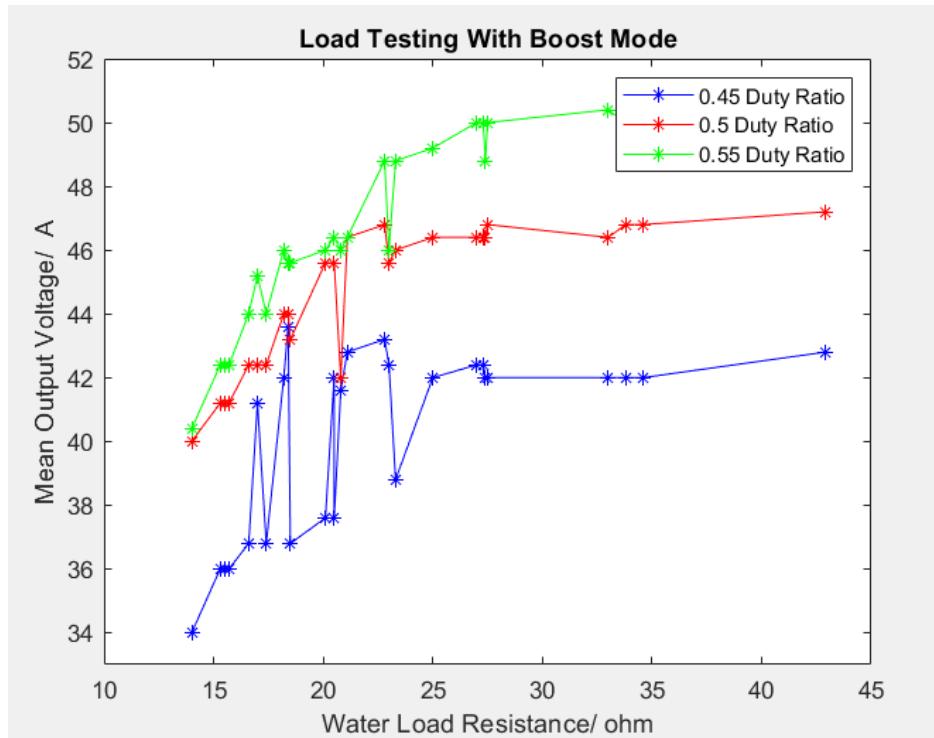


Figure 10.56: Variation of Output Voltage with the Water Load resistance

It showed that with higher duty ratio, the output voltage is higher. And also for loading conditions more than 28ohms, the output voltage becomes steady for all of the test cases.

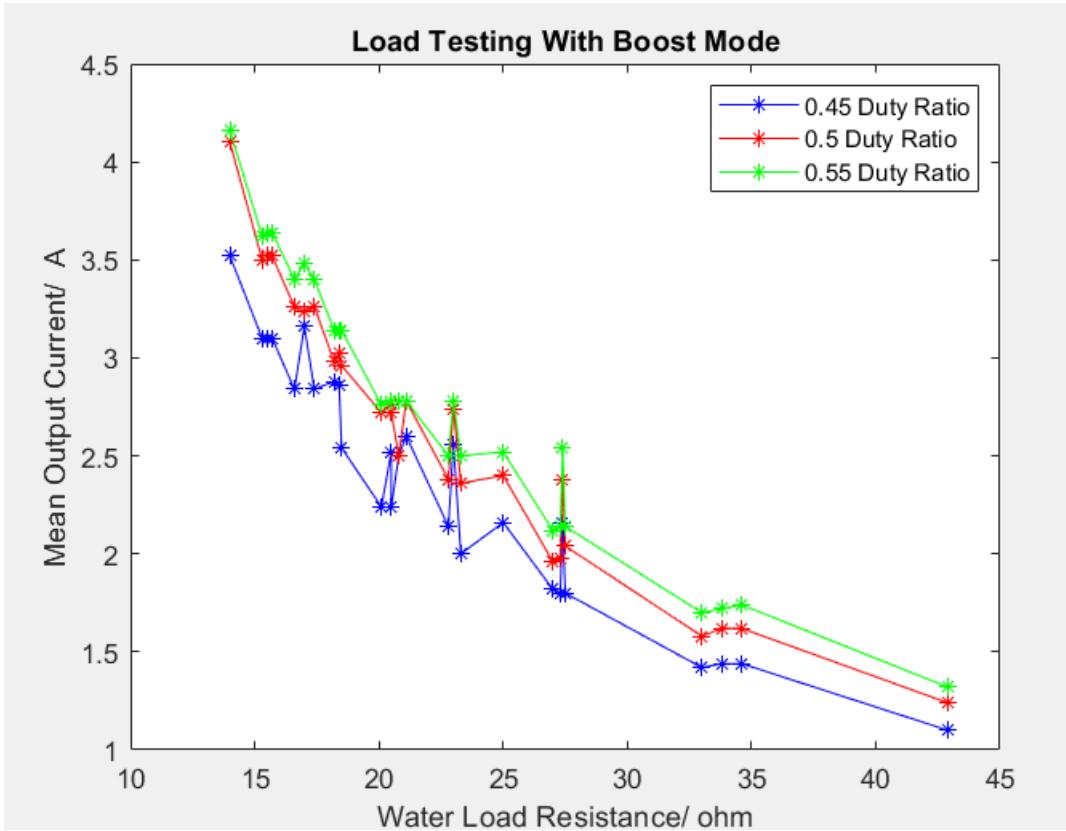


Figure 10.57: Variation of Output current with the Water Load resistance

From the observation of output current, it was observed that with higher loading conditions the current supply becomes lower. And with lower duty ratios output current is much less compared to the higher duty ratio.

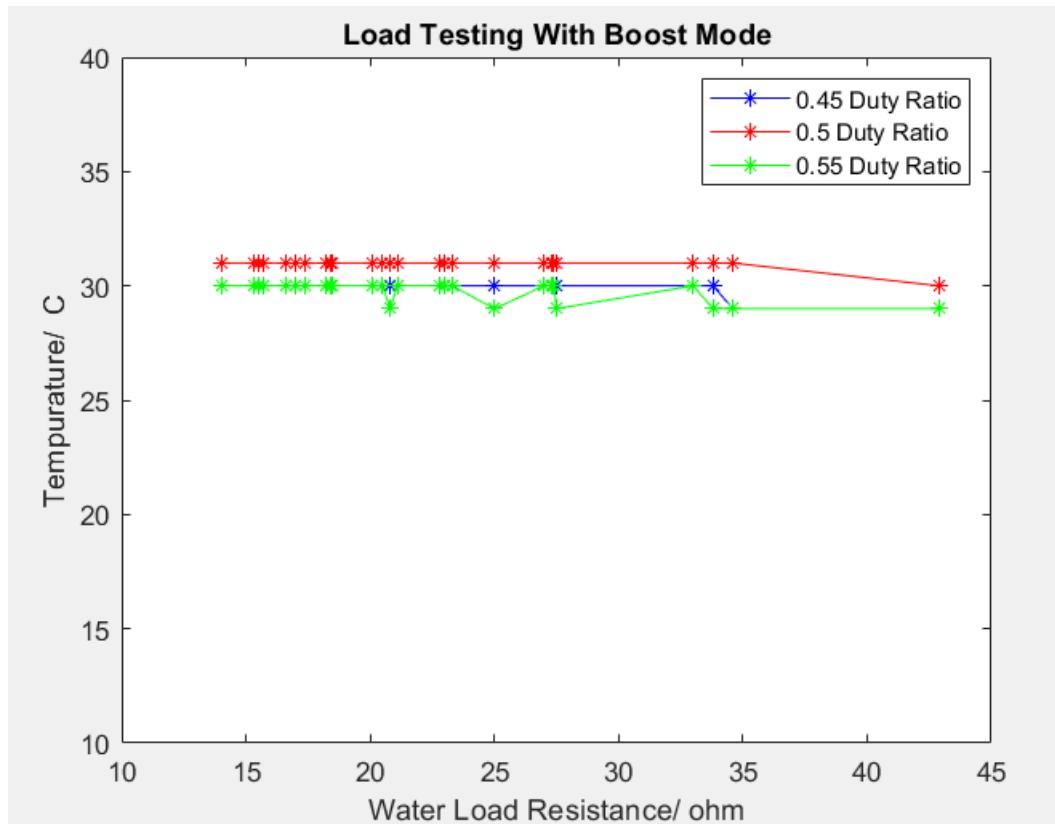


Figure 10.58: Variation of Temperature with the Water Load resistance

10.5 CONCLUSIONS

It was observed that the temperature of the setup remains constant for both fan on and fan off cases. And the setup also consists of a thermostat switch which open circuits when it senses temperature of the heat sink rises more than certain temperature. So as a protective measure the power to the controller can be rerouted through the thermostat so that the device will automatically shutdown if a sudden temperature rise occurred.

CHAPTER 11: PROBLEMS ENCOUNTERED

- **Finding required materials for the battery swapper in the local market**

Most of the items required for the prototype building were either not found in the local market, or too expensive. Therefore, in some cases alternative parts were used to build the swapper prototype

- **Lack of statistical data**

Since the battery swapper based IoT platform is not implemented commercially/Practically, real time data for test the demand prediction and optimization was not available. Therefore, we generated random data set for test the calculations. The random dataset was generated taking into consideration the hourly vehicle traffic in an urban area

- **Compatibility issues with the application development platforms**

At first the initial attempt was to make the mobile application using IONIC platform since it is a hybrid mobile application platform. But when the EKF has to be implemented inside the application the external libraries were not enough since IONIC is a newer platform. Therefore, Android Studio was selected and implemented the app again.

CHAPTER 12: CONCLUSIONS

With the designed and practically implemented battery swapping mechanism, and the IoT platform and with the help of optimization algorithm; it is possible to implement a complete electric vehicle eco system that can be commercially deployable with few minor modifications.

With the generation of real data from an implemented swapping station in the future, the algorithms and the demand prediction methods can be further refined to suit the business needs.

As mentioned in chapter 3.5, the swapping mechanism can be further modified by adding the supercapacitor modules also into the battery swapping mechanism. This way all the power sources and the converters would be integrated into single unit saving space in the electric vehicle.

As per the android application, the google maps-based navigation option can be added to the developed android application (requires payment) so that the electric vehicle user can find the nearest battery swapping station easily.

It is also possible to implement a mutual battery swapping scheme (if electric vehicle battery drained on the road) or, battery delivery scheme (for a business model) ,if the electric vehicle user requests a battery via his android application.

REFERENCES

- [1] C. A. Ellis and S. A. Parbery, “Is smarter better? A comparison of adaptive, and simple moving average trading strategies,” *Research in International Business and Finance*, vol. 19, no. 3, pp. 399–411, Sep. 2005.
- [2] S. Hansun and M. B. Kristanda, “Performance analysis of conventional moving average methods in forex forecasting,” in *2017 International Conference on Smart Cities, Automation Intelligent Computing Systems (ICON-SONICS)*, 2017, pp. 11–17.
- [3] M. R. Sheldon, *Introduction to Probability and Statistics for Engineers and Scientists, Third Edition*, 3rd ed. Academic Press, 2004.
- [4] I. GAVRIIL, G. GRIVAS, A. CHALOULAKOU, N. SPYRELLIS, and P. Kassomenos, “An application of theoretical probability distributions, to the study of PM10 and PM2.5 time series in Athens, Greece,” *Global NEST*, vol. 8, no. 3, pp. 241–251, 2006.
- [5] R. J. Rossi, *Mathematical Statistics: An Introduction to Likelihood Based Inference*, 1st ed. Wiley, 2018.
- [6] “Forecasting of demand using ARIMA model - Jamal Fattah, Latifa Ezzine, Zineb Aman, Haj El Moussami, Abdeslam Lachhab, 2018.” [Online]. Available: <https://journals.sagepub.com/doi/full/10.1177/1847979018808673>. [Accessed: 02-Sep-2019].
- [7] S. G. Makridakis, S. C. Wheelwright, and R. J. Hyndman, *Forecasting: Methods and Applications*, 3rd ed. Wiley, 1997. pp314
- [8] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*, 5th ed. Wiley, 2015.
- [9] D. H. Levenbach, *Change & Chance Embraced: Achieving Agility with Smarter Forecasting in the Supply Chain*. Delphus Publishing, 2017.