

HYBRID CHARGING STATION FOR ELECTRIC VEHICLE

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Electrical and electronics

By

KEERTHANA SAGAR S (38140020)

MUGILARASI Ko Si (38140031)



DEPARTMENT OF ELECTRICAL AND ELECTRONICS

SCHOOL OF ELECTRICAL AND ELECTRONICS

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY- 2022



SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

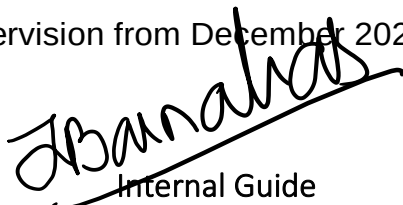
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF ELECTRICAL AND ELECTRONICS

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KEERTHANA SAGAR S (38140020)** and **MUGILARASI Ko Si (38140031)** who carried out the project entitled "**HYBRID CHARGING STATION FOR ELECTRIC VEHICLE**" under my supervision from December 2021 to May 2022.



Internal Guide

Mr. J. Barnabas Paul Glady

Head of the Department

Dr.V. Sivachidambaranathan

Submitted for Viva voca Examination held on 06/05/2022

Internal Examiner

External Examiner

DECLARATION

We, **KEERTHANA SAGAR S (38140020)** and **MUGILARASI Ko Si (38140031)**, hereby declare that the Project Report entitled “**HYBRID CHARGING STATION FOR ELECTRIC VEHICLE**” done by us under the guidance of **Mr. J. Barnabas Paul Glady**. Is submitted in partial fulfillment of the requirements for the award of a Bachelor of Engineering degree in Electrical and Electronics

DATE: 14/03/2022

1.



PLACE: Chennai

2.



**SIGNATURE OF THE
CANDIDATE**

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

We convey our thanks to **Dr. N. M. Nandhita., Dean, School of Electrical and Electronics,** and **Dr. V. Sivachidambaranathan, Head of the Department, Dept. of Electrical and Electronics Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide Mr. J. Barnabas Paul Gladly for her valuable guidance, suggestions and constant encouragement that paved way for the successful completion of my project work.

we wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electrical and Electronics who were helpful in many ways for the completion of the project.

HYBRID SATHYABAMA DC CHARGING STATION FOR ELECTRIC VEHICLE USING IOT

ABSTRACT

Hybrid Grid Charging Stations (HGCS) play a significant role in the adoption of electric vehicles (EV) at remote locations and reduce grid burden in urban areas. The HGCS seeks energy from both renewable and non-renewable sources (RES). Of all the RES, photovoltaic (PV) is the best choice for clean energy due to its abundance and ease of installation. Nevertheless, PV energy fluctuates due to changes in irradiance, and it cannot generate constant energy. To meet the energy demand, an energy storage device and non-renewable source are needed. The proposed system includes a PV array coupled with a boost converter, a non-renewable AC source, and an EV charger. PV energy generated in the absence of or with reduced sunlight is not adequate to meet the demand. EV can therefore meet the demand. As generation exceeds demand, EV utilizes the excess clean energy to stabilize the system. With IoT, EV charging will be improved and the impact will be seen. Transportation systems can benefit from this method. This work is to create an application that connects to the system, displays the available power sources, and displays the power delivered from those sources. This method yielded an off-grid, pollution-free, efficient and reliable EV charging station.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	LIST OF FIGURES	xi
	LIST OF TABLES	xiii
	LIST OF SYMBOLS AND ABBREVIATIONS	xiv
1	INTRODUCTION	
	1.1 INTRODUCTION	01
	1.2 OBJECTIVE	02
	1.3 LITERATURE SURVEY PAPER	03
2	PRINCIPAL OF OPERATION	
	2.1 EXISTING SYSTEM	05
	2.1.1 EXISTING SYSTEM WITH DRAWBACKS	05
	2.2 PROPOSED SYSTEM	07
	2.2.1 BLOCK DIAGRAM	08
3	SIMULATION CIRCUIT AND OUTPUT	
	3.1 MATLAB/SIMULINK	09
	3.1.1 INTRODUCTION	09
	3.1.2 THE LANGUAGE OF TECHNICAL COMPUTING	09
	3.1.3 MATH. GRAPHICS. PROGRAMMING	09
	3.1.4 SCALE. INTEGRATE. DEPLOY	10
	3.1.5 KEY FEATURES	10
	3.1.6 INTERFACING WITH OTHER LANGUAGES	10

3.1.7 WHY MATLAB	11
3.1.8 MATLAB SPEAKS MATH	11
3.1.9 MATLAB IS DESIGNED FOR ENGINEERS AND SCIENTISTS	12
3.1.10 MATLAB INTEGRATE WORKFLOWS	13
3.2 DC-DC CONVERTER (SEPI CONVERTER IN MPPT ALGORITHM)	14
3.3 AC-DC CONVERTER (BRIDGELESS RECTIFIER)	15
3.4 EMBEDDED C	16
3.4.1 ARDUINO	16
3.4.2 SOURCE CODE	18
4 HARDWARE DESCRIPTION	
4.1 TEMPERATURE AND HUMIDITY SENSOR (DHT11)	21
4.1.1 WORKING PRINCIPLE OF DHT11 SENSOR	21
4.1.2 APPLICATIONS	22
4.2 IOT MODULE-NODEMCU (ESP8266)	23
4.2.1 ESP8266 SPECIFICATIONS	23
4.2.2 ESP8266 FUNCTIONS	24
4.2.3 ESP8266 APPLICATIONS	24
4.3 LCD	25
4.3.1 COMMAND REGISTER	28
4.3.2 DATA REGISTER	28
4.4 SOLAR PANEL	28
4.4.1 PANEL RATINGS	28

4.4.2 WORKING PRINCIPLE	29
4.4.3 HOW SOLAR POWER WORKS	30
4.4.4 PHOTOVOLTAICS (PV)	30
4.4.5 PHOTOVOTAICS SOLAR POWER SYSTEMS	30
4.5 SEPI CONVERTER USING MPPT ALGORITHM	31
4.5.1 SEPI CONVERTER	31
4.5.2 BASIC SEPIC OPERATION	32
4.5.3 CONTINUOUS CONDUCTION MODE	32
4.5.4 DISCONTINUOUS CONDUCTION MODE	34
4.5.5 DESIGN OF SEPI CONVERTER	35
4.5.6 MAXIMUM POWER POINT TRACKING	36
4.6 AC-DC CONVERTER USING BRIDGELESS SEPIC	37
4.6.1 MODES OF OPERATION	38
4.7 PROTECTION CIRCUIT	40
4.7.1 EXTERNAL AMPLIFIER CIRCUIT	43
4.8 MICROCONTROLLER	45
4.8.1 DSPIC30F2010	45
4.8.2 HIGH PERFORMANCE MODIFIED RISC CPU	45
4.8.3 DSP ENGINE FEATURES	46
4.8.4 MOTOR CONTROL PWM MODULE FEATURES	46
4.8.5 QUADRATURE ENCODER INTERFACE MODULE FEATURES	47

4.8.6 PROGRAMMABLE BROWN-OUT DETECTION AND RESET GENERATION SPECIAL MICROCONTROLLER FEATURES	47
4.8.7 BLOCK DIAGRAM	48
4.8.8 PIN DESCRIPTION	49
4.9 CORE OVERVIEW	50
4.9.1 PROGRAMMER'S MODEL	52
4.9.2 SOFTWARE STACK POINTER/FRAME POINTER	53
4.9.3 PROGRAM COUNTER	54
4.9.4 MULTIPLIER	55
4.9.5 DATA ACCUMULATORS AND ADDER/SUBTRACTOR	55
4.9.6 ADDER/SUBTRACTOR, OVERFLOW AND SATURATION	56
4.9.7 DATA SPACE WRITE SATURATION	59
4.9.8 BARREL SHIFTER	59
4.10 CALCULATION	60
4.10.1 E-BICYCLE BATTERY	60
4.10.2 PANEL	60
4.10.3 BRIDGELESS RECTIFIER	61
4.11 EXPERIMENTAL OUTPUT	61
4.11.1 CHECKING THE WEATHER CONDITION TO CHOOSE SOURCE OF CHARGING	61

4.11.2 PV CONNECTION	62
4.11.3 CHARGING	64
4.11.4 GRID CONNECTION	65
4.11.5 CHARGING	67
4.12 FEATURES	68
5 CONCLUSION	
5.1 CONCLUSION	69
REFERENCES	70

LIST OF FIGURES

FIGURE NO NO	TITLE	PAGE
2.1	Existing charging station model (for E-car charging)	05
2.2	Block diagram of proposed system	08
3.1	SEPI converter matlab model	14
3.2	Output waveform of SEPI converter (V/I curve)	14
3.3	Bridgeless rectifier matlab model	15
3.4	Output waveform of bridgeless rectifier (V/I curve)	15
3.5	Program for microcontroller in Embedded C using Arduino	20
4.1	Temperature and humidity sensor	21
4.2	DHT11 sensor	22
4.3	IOT ESP8266	23
4.4	LCD 16*2	26
4.5	LCD-16*2-Pin diagram	27
4.6	Panel designing model	29
4.7	Working diagram of PV system with SEPIC converter	31
4.8	SEPI Converter	32
4.9	Switch ON condition	33
4.10	Switch OFF condition	34
4.11	Power/Voltage curve	36
4.12	Basic P&O algorithm flow chart	37
4.13	Bridgeless rectifier	38
4.14	Modes of operation of rectifier	38
4.15	Topologies of rectifier	39
4.16	Filter circuit	43
4.17	Pin diagram of OP-AMP	44
4.18	OP-AMP design	44

4.19	Pin diagram of microcontroller	45
4.20	Block diagram of microcontroller	48
4.21	Blynk notification	62
4.22	PV charging circuit	62
4.23	Output of panel at maximum radiation	63
4.24	PV output at ground floor	63
4.25	E-bicycle displayed in half charged	64
4.26	E-bicycle charged through PV charging circuit	64
4.27	Blynk notification	65
4.28	Grid charging circuit	66
4.29	E-bicycle charged by grid	67
4.30	Half charged by grid displayed by E-bicycle	67

LIST OF TABLES

1.1 Classification of stages according to switch, diodes, inductor current	39
--	----

LIST OF SYMBOLS AND ABBREVIATIONS

Ω	-	Ohm
V in	-	Input Voltage
V out	-	Output Voltage
Z	-	Impedance
mA	-	Milli Ampere
μ s	-	Micro Seconds
mm	-	Milli Meter
KWH	-	Kilo Watt Hour
MHz	-	Mega Hertz
DC	-	Direct Current
AC	-	Alternating Current
PV	-	Photo Voltaic
PWM	-	Pulse Width Modulation
RPM	-	Revolution per Minute
MPPT	-	Maximum Power Point Tracking
DCR	-	DC Resistance

CHAPTER -1

INTRODUCTION

1.1 INTRODUCTION:

According to the current Indian scenario, the demand for electric vehicles (EV) is growing exponentially. Based on India's level of atmospheric pollution, the statement is valid. Choudhary (1997) concludes that India consumes most fossil fuels for transportation purposes. It is predicted that the number of fossil fuels will decline significantly in the future. In light of this, EVs are necessary for the automobile industry today. The grid powers these EVs, which is what is referred to as a grid to vehicle (G2V). Ketton and Letendre (1997) introduced the concept of Vehicle to Grid (V2G) in which the energy stored in an electric vehicle battery could be used to generate electricity for the grid. The battery storage may also be used to regulate the frequency of energy output when the EV is transferring electricity to the grid. Alternatively, the battery side is used as a boost converter, and the grid side is used as an inverter in V2G mode. Pollution-free transportation requires electric vehicles. Therefore, it is necessary to shift from grid-based EV charging stations to off-grid charging stations (OGCS). Singh et al. (2016) proposes that the source for OGCS is renewable energy. At remote locations where the grid is not accessible, it can be used. Photovoltaic (PV) and wind are the RESs used by the OGCS. To generate power, however, wind energy requires more conversion stages than PV. Therefore, the feasibility of a PV energy-based off-grid charging station is more. A PV-based EV charging station has been presented by Bhatti and Salam (2016). This is a result of abundant solar energy and the ease of installation of PV power generation systems. The PV charging station has been supported by several articles. At the parking garage, Tupulet al. (2013) recommended a PV-based CS that was cost-effective. Parking rates, installation costs, and tax rebates are used to calculate the cost of the system. Both garages and vehicle owners benefit from this. Along with this benefit, the system reduces the grid's charge burden and promotes the use of renewable energy in transportation. In terms of per unit cost and installation cost, the authors have not presented an economic analysis of this methodology for other locations. Further, the system is PV-dependent, which is unsustainable because irradiance is intermittent. Solar charging stations with the grid were presented by Goli

and Shireen (2014). When PV energy is insufficient for EVs, the grid is supporting the CS. This limits the system's sustainability since the grid isn't available everywhere. As a related example, Hernandez and Sutil (2016) presented a PV-based CS supporting the system framework during peak loads, including regenerative braking. According to the author, he aims to reduce the charging costs by utilizing as much RES as possible. Similarly, Li et al. (2013) proposed a PV and wind-based CS with battery storage to handle fluctuations in power generation under variable environmental conditions. An SoC-based approach is presented in this paper to overcome the problems caused by RES. However, they failed to determine the ideal SoC estimation strategy for EV batteries. According to the literature mentioned above, the present scenario calls for a self-generated / off-grid CS. Although, the rapid increase of EVs imposes a burden on the grid. The successful implementation of the EV will require RES-based charging stations, especially on the highways and at remote locations. Therefore, a PV-based off-grid CS is proposed. Among the possible options listed in the literature, the inclusion of battery storage is a suitable approach for making the proposed system sustainable. During variable irradiance conditions, a PV source coordinated with ESS is an efficient source of energy. Additionally, ESS utilizes the maximum amount of available renewable energy whenever there is surplus power or no EVs are available. In this way, an off-grid CS is eco-friendly and efficient.

In accordance with the above theories and their valid outcomes, we have proposed a hybrid charging station. We have utilized various energy resources such as solar power, grid power for inclement weather, as well as manual grid shifting and IoT modules for live monitoring in this hybrid charging station.

1.2 OBJECTIVE:

- The aim of this work is to develop a hybrid grid consisting of a renewable source and a non-renewable source as its supply.
- Monitoring the requirements, usage and losses of input supply
- Backing up with storage systems for peak demands

1.3 LITERATURE SURVEY

Alper Kerem, et. al.,(2017) designed a Smart grid integration of micro hybrid power system using 6-switched 3-level inverter from IEEE International Istanbul smart grid Congress and Fair, have proposed a hybrid power system modeled in Matlab/Simulink environment is presented in this study. This model includes a hybrid system consisting of wind and solar panels and it is connected to the smart grid through a 6-switched 3-level inverter. A 6-switched 3-level inverter is preferred in this study as an inverter due to its success in minimizing the harmonics. The signals required for the operation of the semiconductor switches within the inverter are obtained by the space-vector pulse width modulation technique. The system fed the receivers by generating electricity energy and sent the generated excess energy to the grid (grid) on favorable conditions, that is, when wind and/or sun are sufficient for energy production. The receivers are fed by the grid on unfavorable conditions. By means of analyzing the performance of the system, it is determined that the system is stable.

Shireen.W, et. al .,(2010) invented Plug-in Hybrid Electric vehicles in the smart grid environment. IEEE PES transmission and distribution conference and exposition, and have proposed Plug-in hybrid electric vehicles (PHEVs) have the potential to curb emission and reduce the cost of transportation. Another unique advantage of PHEVs is their capability to integrate the onboard energy storage units with the power grid which can improve the efficiency and increase the reliability of the power grid. These concepts are still new and under development. Unlike Hybrid Electric vehicles (HEVs) a major concern in the case of PHEVs is the impact on the grid when they are plugged in for charging to the already loaded grid. In view of this, this paper presents the trends in analysis, design and evaluation of PHEVs in the future smart grid environment.

Harun Turker, et. al .,(2018) invented Multiobjective optimization of Grid- Photovoltaic- Electric Vehicle Hybrid system in Smart Building with Vehicle-to-Grid (V2G) concept, IEEE International Conference on Renewable Energy Research and Applications, have proposed this paper presents an optimal Energy Management Strategy (EMS) to optimize Grid /PV /EV hybrid system in Smart Building with a

Vehicle-to-Grid (V2G) concept and so bidirectional capability of the vehicle. The aims of the multi objective optimization are to maximize the selfconsumption of the solar production and to optimal charging the vehicle together. Linear Programming method is used to solve the optimization problem. The use cases results show the great economic profitability potential of the studied system.

Bibaswan Bose, et. al., (2020) Multi-loop Multi-objective Control of Solar Hybrid EV Charging Infrastructure for Workplace, IEEE International conference on Advances in computing, communication control and networking, have proposed the increasing demand of EV has led to an increase in need of reliable charging infrastructure which can be integrated with the conventional grid. The deployment of smart grid projects has resulted in modernization of conventional grid and evolving it into a more reliable form by integrating renewable sources and EV into the same infrastructure. This work proposes to use SPVA-battery-grid based EV charging infrastructure which will increase the reliability on the system and provide instantaneous supply of power. Further, this work presents case scenario of workspace which operates from 9am to 5pm. The MATLAB simulation shows that proposed power flow control logic can successfully switch between available multiple modes and supply demanded power to EV.

CHAPTER-2

PRINCIPAL OF OPERATION

2.1 EXISTING SYSTEM:

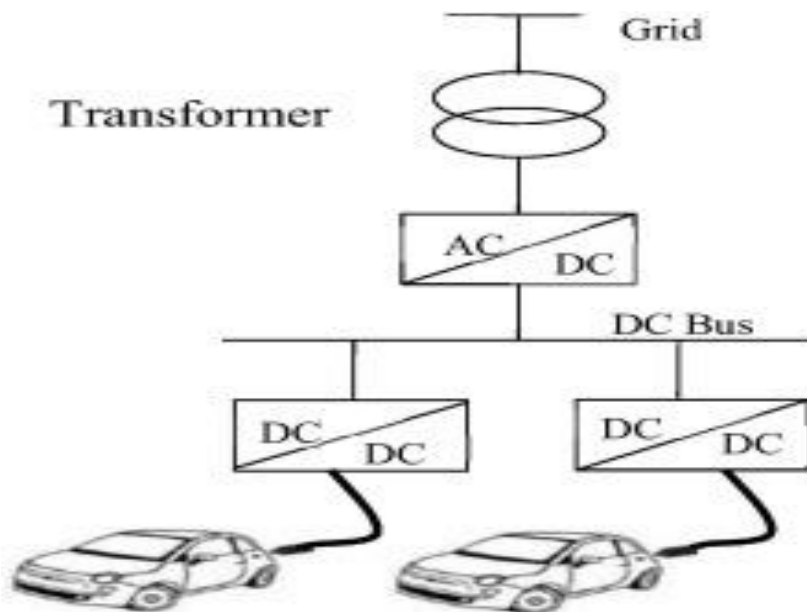


Fig 2.1 Existing charging station model (for E-car charging)

Figure 2.1 shows the existing charging station for electric car charging. In this there is a grid connected to the step down transformer and then to the AC-DC converter used to convert ac to dc to charge the electric vehicle via DC-DC converter (Boost or Buck converter).

2.1.1 EXISTING SYSTEM WITH DRAWBACKS:

Since the last decade, the number of electric vehicles (EVs) has significantly increased. This is due to the new advanced technology especially in electric motor design and also the government policies to reduce the amount of combustion engine produced carbon footprint into the atmosphere. As a result, more and more countries around the world such as United Kingdom, has taken the initiative to reduce the

dependency on the usage of internal combustion engine vehicles to combat this problem. Due to the reasons mentioned earlier, EV's will soon be the main focus in transportation changes. Moreover, most of the EV aspects such as the charging/discharging of the battery, battery energy management for driving the electric motor or vice versa, the concepts of Vehicle to Grid (V2G) or Grid to Vehicle (G2V) are well studied by the researchers around the world. However, less focus has been given to the issues related to charging stations.

As known, the EV charging station consists only a power converter topology. Then, it is connected to the existing electrical grid using a rectifier and a DC converter to transfer the energy between the EV to the existing electrical grid or vice versa, but it can always remain the stability of the processes. As an example, when the EV is connected to the charging station, the behaviour of the charging station will change accordingly to the converter operation either rectification or inverter mode to the EV. Therefore, the EV sees the power converter as a nonlinear input source during rectification mode. In the meantime, the EV battery will actively contribute to the frequency variation/disturbance of the power grid system if not be controlled. In this case, if the grid is continuously disturbed, the power flows between the grid and the battery cannot be consistent for providing a fast charging condition for the EV. This problem does not only affect the power flows but also it reduces the power factor and grid frequency at the grid supply. Therefore, it is important for the EV to behave as a continuous primary frequency control to the grid side that will not interrupt the stability of the grid. Meanwhile, as seen from inside the EV perspective, the ability to give a high response to the state of charge (SOC) of the battery is also essential. However now a consider to any harmonic injection due to line impedance to the EV battery rectification mode is also needs to be considered.

For example, one EV charging station does not create a significant problem, but when hundreds of charging stations are connected at the same Point of Common Coupling (PCC), a major issue is created. Here, it will impose a tremendous problem on the grid stability due to frequency fluctuation and also reducing the power factor at the PCC. Therefore, it is necessary to make sure that the EV charging station can minimize these impacts to the electric grid by providing a high quality power factor, flexible frequency restoration and while the EV battery can be charged at the very short of time. Several researchers have suggested using two-stage power converters

or two-stage controllers. The two stages power converters are made by combining the converter (rectifier) with a DC converter before it connects to the EV in order to maintain the SOC of the battery during charging, while maintaining the voltage for electrical grid quality. It can also improve the life cycle of the battery before it's been saturated. Moreover, this configuration needs dual loop controllers for the converters for rectifier: first it needs to maintain the DC link voltage at the DC converter input using outer voltage loop control and then it needs to have a current flow control as an inner loop control to maintain the power factor at the AC grid level. Due to the complexity of the controllers which has to maintain the stability gain and two converters circuit at the same time. The best solution is to use only one power converter and one control loop where it should be able to keep the voltage charging at the EV at a constant rate and at the same time control the quality of the grid current. The idea is to implement the synchronous motor converter control formulation suggested in or any voltage control with has faster power quality solution and frequency restoration. By applying this idea, it is predicted that EV source and electrical grid will benefit from a good quality power while at the same time the EV will be the primary driver of next power generation to the electrical system for the future.

2.2 PROPOSED SYSTEM:

Hybrid grid charging station (HGCS) plays a significant role. The HGCS seeks energy from renewable and non-renewable energy sources (RES). Amongst all RES, the photovoltaic (PV) is the best suitable clean energy source due to abundance and simple installation. However, PV energy fluctuates due to change in irradiance and it cannot generate the constant energy. Therefore, an energy storage device and non-renewable source is required to meet the energy demand and improve the sustainability of the charging station.

Thereby, a system has been proposed which consists of an energy storage system EV along with the PV source, a non-renewable AC source and a EV charger. In this system the solar panel and grid for power supply to charging the vehicle lithium-ion battery. The solar panel produce the DC supply. DC voltage is controlled by SEPI converter. The AC voltage getting from the grid, which is converter into DC controlled voltage by using bridgeless rectifier.

Check the availability in the solar panel using temperature sensor, If the required power, temperature to charge the battery is available in the solar source, using the source of solar to charge the battery. The energy generated from PV is not sufficient to meet the demand during the absence of or reduced sunlight, using the grid to charge the battery. The available source displayed to the user. And also using the productive circuit when the solar panel are not used to charge

2.2.1 BLOCK DIAGRAM:

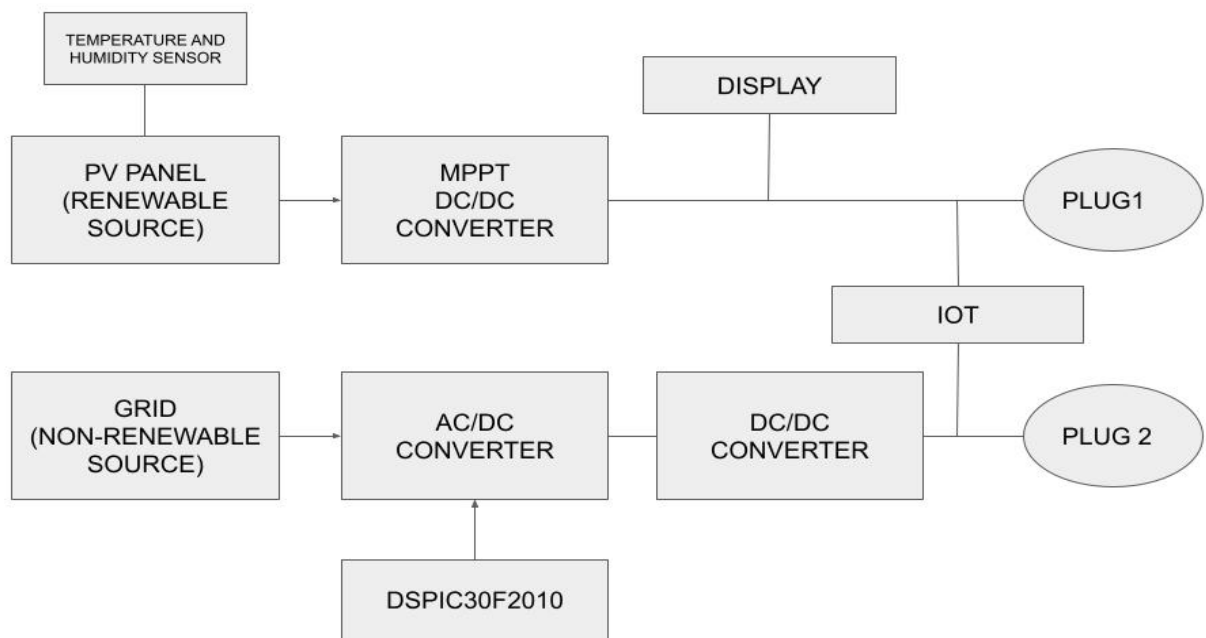


Fig 2.2 Block diagram of proposed system

Figure 2.2 shows the block diagram of the proposed system, we are using 2 plug for charging. In plug 1 the input is fed from PV panel to DC-DC converter. It steps up the voltage and charge the electric vehicle, in this we are using IoT and temperature and humidity sensor to display which plug can be used to charge. In plug 2 the input is fed from direct grid and then it is converted into DC by AC-DC converter and then used to charge the electric vehicle. We are using a microcontroller for controlling the flow of current in both the plugs separately.

CHAPTER -3

SIMULATION CIRCUIT AND OUTPUT

3.1 MATLAB/SIMULINK

3.1.1 INTRODUCTION

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment and fourth-generation programming language. A proprietary programming language developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, C#, Java, Fortran and Python.

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing abilities. An additional package, Simulink, adds graphical multi-domain simulation and model-based design for dynamic and embedded systems.

In 2004, MATLAB had around one million users across industry and academia. MATLAB users come from various backgrounds of engineering, science, and economics.

3.1.2 The Language of Technical Computing

Millions of engineers and scientists worldwide use MATLAB® to analyze and design the systems and products transforming our world. MATLAB is in automobile active safety systems, interplanetary spacecraft, and health monitoring devices, smart power grids, and LTE cellular networks. It is used for machine learning, signal processing, image processing, computer vision, communications, computational finance, control design, robotics, and much more.

3.1.3 Math. Graphics. Programming.

The MATLAB platform is optimized for solving engineering and scientific problems. The matrix-based MATLAB language is the world's most natural way to express computational mathematics. Built-in graphics make it easy to visualize and gain insights from data. A vast library of prebuilt toolboxes lets you get started right away with algorithms essential to your domain. The desktop environment invites experimentation, exploration, and discovery. These MATLAB tools and capabilities are all rigorously tested and designed to work together.

3.1.4 Scale. Integrate. Deploy.

MATLAB helps you take your ideas beyond the desktop. You can run your analyses on larger data sets and scale up to clusters and clouds. MATLAB code can be integrated with other languages, enabling you to deploy algorithms and applications within web, enterprise, and production systems.

3.1.5 Key Features

- High-level language for scientific and engineering computing
- Desktop environment tuned for iterative exploration, design, and problem-solving
- Graphics for visualizing data and tools for creating custom plots
- Apps for curve fitting, data classification, signal analysis, and many other domain-specific tasks
- Add-on toolboxes for a wide range of engineering and scientific applications
- Tools for building applications with custom user interfaces
- Interfaces to C/C++, Java, .NET, Python, SQL, Hadoop, and Microsoft Excel
- Royalty-free deployment options for sharing MATLAB programs with end users

3.1.6 Interfacing with other languages

MATLAB can call functions and subroutines written in the programming languages C or Fortran. A wrapper function is created allowing MATLAB data types to be passed and returned. The dynamically loadable object files created by

compiling such functions are termed "MEX-files" (for **MATLAB executable**). Since 2014 increasing two-way interfacing with Python is being added.

Libraries written in Perl, Java, ActiveX or .NET can be directly called from MATLAB and many MATLAB libraries (for example XML or SQL support) are implemented as wrappers around Java or ActiveX libraries. Calling MATLAB from Java is more complicated, but can be done with a MATLAB toolbox which is sold separately by MathWorks, or using an undocumented mechanism called JMI (Java-to-MATLAB Interface), (which should not be confused with the unrelated Java Metadata Interface that is also called JMI). Official MATLAB API for Java was added in 2016.

As alternatives to the MuPAD based Symbolic Math Toolbox available from MathWorks, MATLAB can be connected to Maple or Mathematica. Libraries also exist to import and export MathML.

3.1.7 Why MATLAB?

MATLAB® is the easiest and most productive software for engineers and scientists. Whether you're analyzing data, developing algorithms, or creating models, MATLAB provides an environment that invites exploration and discovery. It combines a high-level language with a desktop environment tuned for iterative engineering and scientific workflows.

3.1.8 MATLAB Speaks Math

The matrix-based MATLAB language is the world's most natural way to express computational mathematics. Linear algebra in MATLAB looks like linear algebra in a textbook. This makes it straightforward to capture the mathematics behind your ideas, which means your code is easier to write, easier to read and understand, and easier to maintain.

You can trust the results of your computations. MATLAB, which has strong roots in the numerical analysis research community, is known for its impeccable numerics. A MathWorks team of 350 engineers continuously verifies quality by running millions of tests on the MATLAB code base every day.

MATLAB does the hard work to ensure your code runs quickly. Math operations are distributed across multiple cores on your computer, library calls are

heavily optimized, and all code is just-in-time compiled. You can run your algorithms in parallel by changing for-loops into parallel for-loops or by changing standard arrays into GPU or distributed arrays. Run parallel algorithms in infinitely scalable public or private clouds with no code changes.

The MATLAB language also provides features of traditional programming languages, including flow control, error handling, object-oriented programming, unit testing, and source control integration.

3.1.9 MATLAB Is Designed for Engineers and Scientists

MATLAB provides a desktop environment tuned for iterative engineering and scientific workflows. Integrated tools support simultaneous exploration of data and programs, letting you evaluate more ideas in less time.

- You can interactively preview, select, and pre-process the data you want to import.
- An extensive set of built-in math functions supports your engineering and scientific analysis.
- 2D and 3D plotting functions enable you to visualize and understand your data and communicate results.
- MATLAB apps allow you to perform common engineering tasks without having to program. Visualize how different algorithms work with your data, and iterate until you've got the results you want.
- The integrated editing and debugging tools let you quickly explore multiple options, refine your analysis, and iterate to an optimal solution.
- You can capture your work as sharable, interactive narratives.

Comprehensive, professional documentation written by engineers and scientists is always at your fingertips to keep you productive. Reliable, real-time technical support staff answers your questions quickly. And you can tap into the knowledge and experience of over 100,000 community members and MathWorks engineers on MATLAB Central, an open exchange for MATLAB and Simulink® users.

MATLAB and add-on toolboxes are integrated with each other and designed to work together. They offer professionally developed, rigorously tested, field-hardened, and fully documented functionality specifically for scientific and engineering applications.

The Live Editor includes results together with the code that produced them to accelerate exploratory programming and analysis. Add equations, images, hyperlinks, and formatted text to enhance your narrative.

3.1.10 MATLAB Integrates Workflows

Major engineering and scientific challenges require broad coordination to take ideas to implementation. Every handoff along the way adds errors and delays.

MATLAB automates the entire path from research through production. You can:

- Build and package custom MATLAB apps and toolboxes to share with other MATLAB users.
- Create standalone executable to share with others who do not have MATLAB.
- Integrate with C/C++, Java, .NET, and Python. Call those languages directly from MATLAB, or package MATLAB algorithms and applications for deployment within web, enterprise, and production systems.
- Convert MATLAB algorithms to C, HDL, and PLC code to run on embedded devices.
- Deploy MATLAB code to run on production Hadoop systems.

MATLAB is also a key part of Model-Based Design, which is used for multidomain simulation, physical and discrete-event simulation, and verification and code generation. Explore Simulink, [Simscap](#), and Stateflow to learn more about Model-Based Design.

3.2 DC-DC CONVERTER (SEPI CONVERTER IN MPPT ALGORITHM)

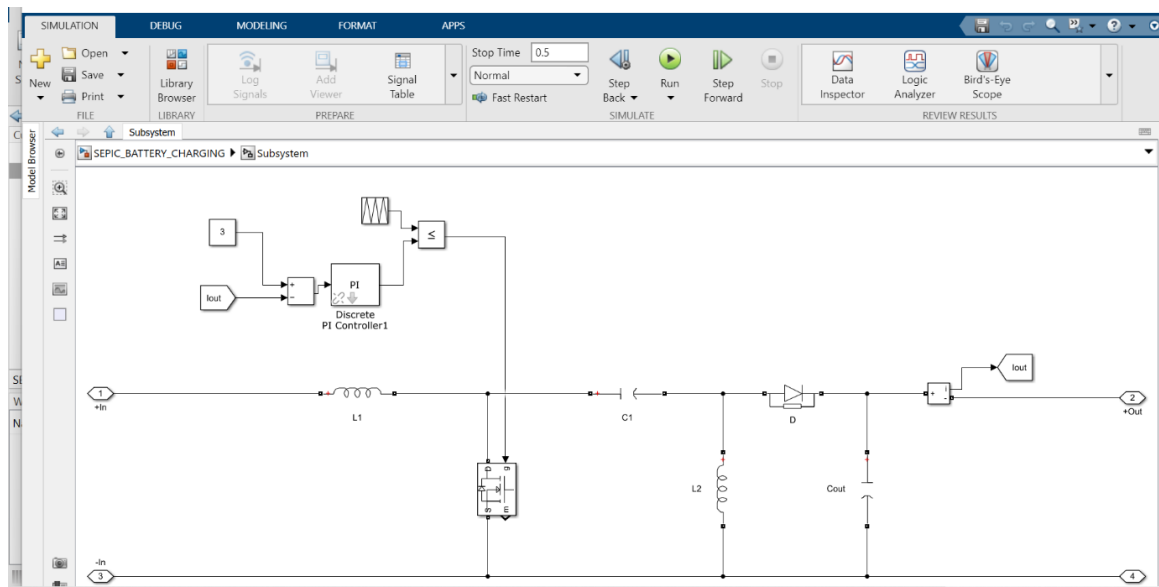


Fig 3.1 SEPI Converter matlab model

Figure 3.1 shows the MATLAB model of SEPI Converter. Input voltage is given as 36V and it will step up as 42V to charge the electric vehicle. We are using MOSFET as a switch to control the flow of current through the circuit. PI controller is used for controlling the FIG whole circuit. Pulse generator is used to give a PWM pulse to the controller via MOSFET and comparator is used for achieving maximum PWM. Inductor and capacitor are used as filtering circuit. Diode is used for flow the current through one way.

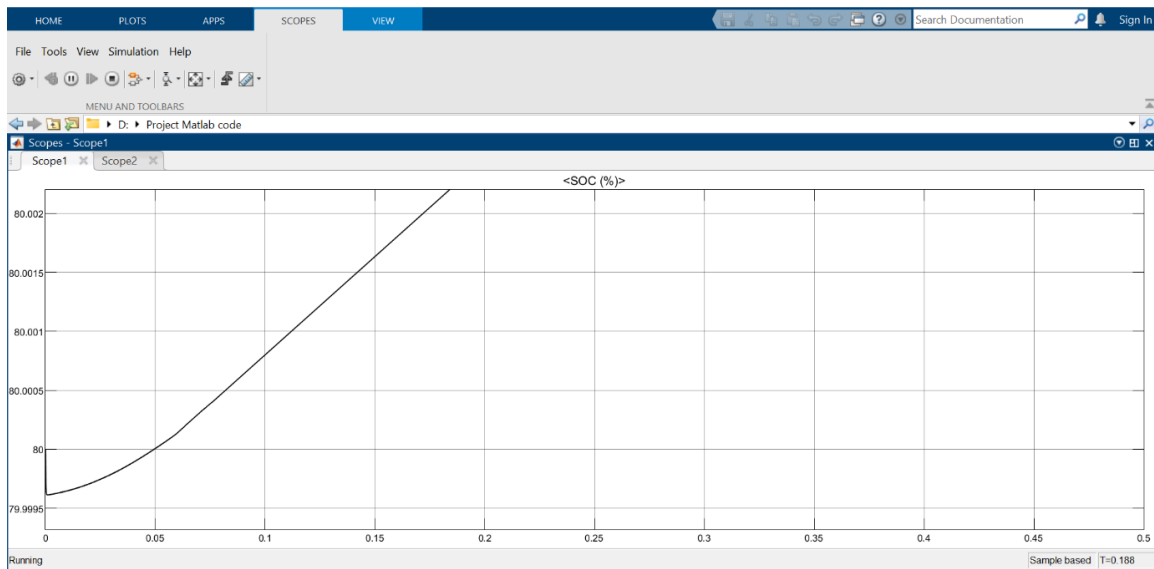


Fig 3.2 Output Waveform of Sepi Converter (V/I Curve)

3.3 AC -DC CONVERTER (BRIDGELESS RECTIFIER)

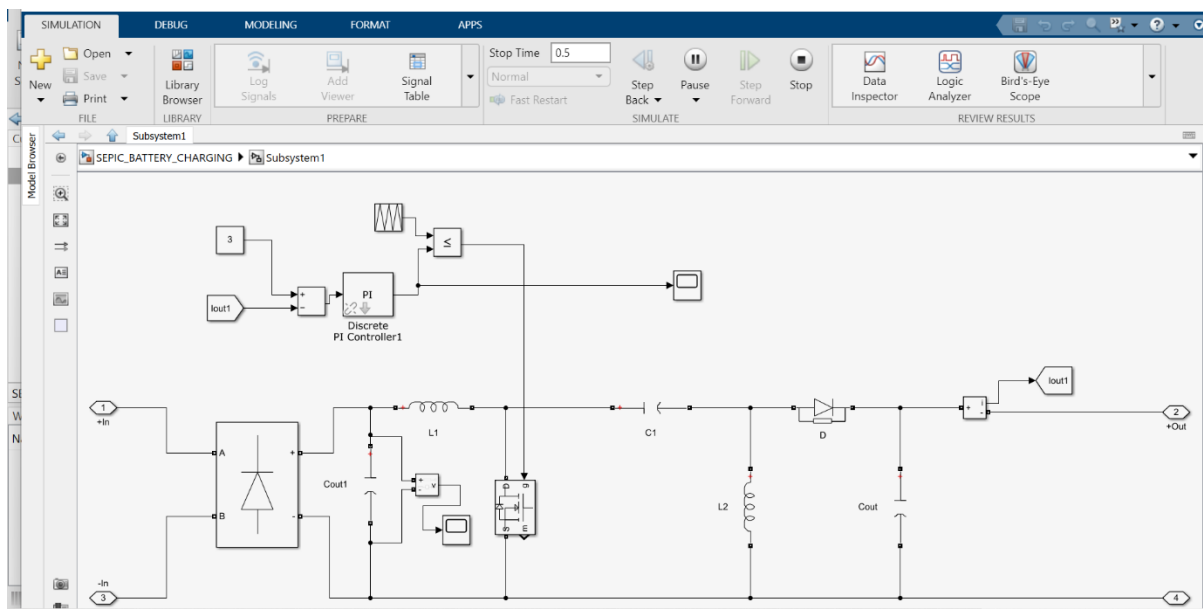


Fig 3.3 Bridgeless Rectifier MATLAB Model

Figure 3.3 shows the matlab model for Bridgeless rectifier, In this input is fed from grid 24v ac and it will convert 42v dc to charge the Electric vehicle. Here we are using MOSFET as a switch controlled by PI controller. Pulse Generator gives the pulse to PI and comparator is used to achieve maximum PWM. Inductor and capacitor are used as filters, diodes are used as switch current flows through one way.

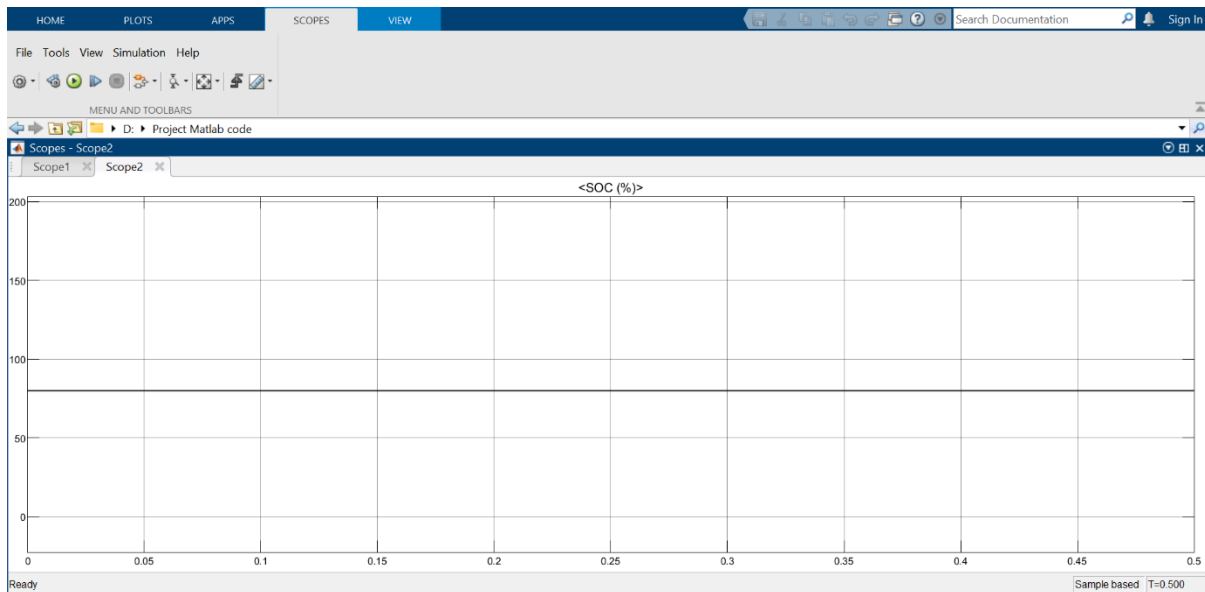


Fig 3.4 Output waveform of bridgeless rectifier(V/I Curve)

Figure 3.4 shows the output waveform of bridgeless rectifier with constant voltage level of 42V and maximum current is 5A.

3.4 EMBEDDED C:

Embedded c is a set of language extension for the C Programming language by the C Standards committee to address commonality issues that exist between C extensions for different embedded systems. Historically embedded C programming requires nonstandard extensions to the C language in order to support exotic features such as fixed-point arithmetic, multiple distinct memory banks, and basic I/O operations. Embedded C uses most of the syntax and semantics of standard C, e.g., main() function, variable definition, data type declaration, conditional statements(if, switch, case), loops(while, for), functions, arrays and strings, structures and union, bit operations, macros, etc. During infancy years of microprocessor-based systems, programs were developed using assemblers and fused into the EPROMs. there used to be no mechanism to find what the program was doing. LEDs, switches, etc. were used to check for correct execution of the program. But they were too costly and were not quite reliable as well .As time progressed, use of microprocessor-specific assembly-only as the programming language reduced and embedded systems moved onto C as the embedded programming language of choice. C is the most widely used programming language for embedded processors/controllers.

3.4.1 ARDUINO

A program for Arduino hardware may be written in any programming language with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit AVR and 32-bit Cortex-M-based microcontrollers: AVR Studio (older) and Atmel Studio (newer).

The Arduino integrated development environment (IDE) is a cross-platform application (for Microsoft Windows, macOS, and Linux) that is written in the Java programming language. It originated from the IDE for the languages *Processing* and *Wiring*. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple *one-click* mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions, and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.

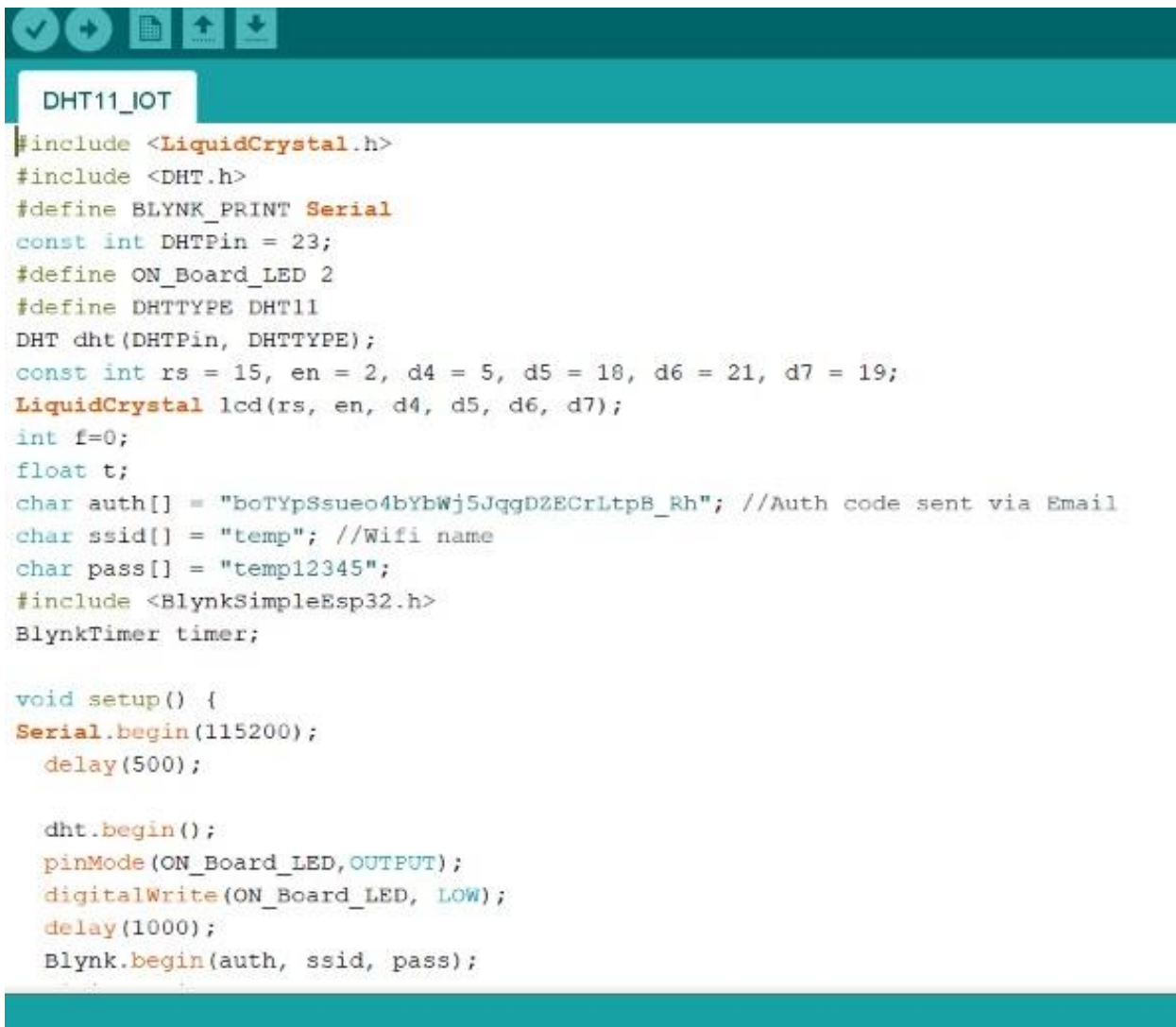
The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program *avrdude* to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

Blynk is the most popular Internet of Things platform for connecting any hardware to the cloud, designing apps to control them, and managing your deployed products at scale.

- With Blynk Library you can connect **over 400 hardware models** (including ESP8266, ESP32, NodeMCU, all Arduinos, Raspberry Pi, Particle, Texas Instruments, etc.) to the Blynk Cloud.
- With Blynk apps for **iOS** and **Android** apps you can easily drag-n-drop graphic interfaces for any DIY or commercial project. It's a pure WYSIWG experience: no coding on iOS or Android required.

- Hardware can connect to Blynk Cloud (open-source server) over the Internet using hardware connectivity available on your board (like ESP32), or with the use of various shields (Ethernet, WiFi, GSM, LTE, etc). Blynk Cloud is available for every user of Blynk **for free**. Direct connection over Bluetooth is also possible

3.4.2 Source Code



```

DHT11_IOT

#include <LiquidCrystal.h>
#include <DHT.h>
#define BLYNK_PRINT Serial
const int DHTPin = 23;
#define ON_Board_LED 2
#define DHTTYPE DHT11
DHT dht(DHTPin, DHTTYPE);
const int rs = 15, en = 2, d4 = 5, d5 = 18, d6 = 21, d7 = 19;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
int f=0;
float t;
char auth[] = "boTYp3sueo4bYbWj5JqgDZECrLtpB_Rh"; //Auth code sent via Email
char ssid[] = "temp"; //Wifi name
char pass[] = "temp12345";
#include <BlynkSimpleEsp32.h>
BlynkTimer timer;

void setup() {
  Serial.begin(115200);
  delay(500);

  dht.begin();
  pinMode(ON_Board_LED, OUTPUT);
  digitalWrite(ON_Board_LED, LOW);
  delay(1000);
  Blynk.begin(auth, ssid, pass);
}

```



DHT11_IOT

```
Blynk.begin(auth, ssid, pass);  
digitalWrite(ON_Board_LED, HIGH);  
timer.setInterval(1000L, getSendData);  
lcd.begin(16, 2);  
delay(1000);  
}  
  
void loop() {  
  if(t>32){  
    lcd.clear();  
    lcd.setCursor(0,0);  
    lcd.print("CONNECT TO");  
    lcd.setCursor(0,1);  
    lcd.print("SOLAR PORT");  
    delay(500);  
  }  
  else{  
    lcd.clear();  
    lcd.setCursor(0,0);  
    lcd.print("CONNECT TO");  
    lcd.setCursor(0,1);  
    lcd.print("GRID PORT");  
    delay(500);  
  }  
  delay(500);  
  Blynk.run();  
}
```



```

DHT11_IOT
11/10/2021

    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("CONNECT TO");
    lcd.setCursor(0,1);
    lcd.print("SOLAR PORT");
    delay(500);
}

else{
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("CONNECT TO");
    lcd.setCursor(0,1);
    lcd.print("GRID PORT");
    delay(500);
}

delay(500);
Blynk.run();
timer.run();
float h = dht.readHumidity(); //--> Read humidity.
t = dht.readTemperature(); //--> Read temperature as Celsius (the default).

// Check if any reads failed and exit early (to try again).
if (isnan(h) || isnan(t)) {
    Serial.println(" Failed to load DHT sensor !");
    delay(1000);
    return;
}

```

Fig 3.5 Program for Microcontroller in Embedded C Using Arduino

CHAPTER - 4

HARDWARE DESCRIPTION

4.1 TEMPERATURE AND HUMIDITY SENSOR (DHT11):

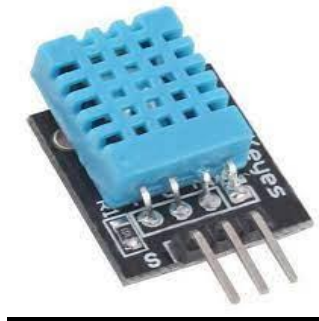


Fig 4.1 Temperature and humidity sensor

Humidity is the measure of water vapour present in the air. The level of humidity in air affects various physical, chemical and biological processes. In industrial applications, humidity can affect the business cost of the products, health and safety of the employees. So, in semiconductor industries and control system industries measurement of humidity is very important. Humidity measurement determines the amount of moisture present in the gas that can be a mixture of water vapour, nitrogen, argon or pure gas etc... Humidity sensors are of two types based on their measurement units. They are a relative humidity sensor and Absolute humidity sensor. DHT11 is a digital temperature and humidity sensor.

DHT11 is a low-cost digital sensor for sensing temperature and humidity. This sensor can be easily interfaced with any micro-controller such as Arduino, Raspberry Pi etc... to measure humidity and temperature instantaneously.

DHT11 humidity and temperature sensor is available as a sensor and as a module. The difference between this sensor and module is the pull-up resistor and a power-on LED. DHT11 is a relative humidity sensor. To measure the surrounding air this sensor uses a thermistor and a capacitive humidity sensor.

4.1.1 Working Principle of DHT11 Sensor

DHT11 sensor consists of a capacitive humidity sensing element and a thermistor for sensing temperature. The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in

the capacitance value occurs with the change in humidity levels. The IC measure, process this changed resistance values and change them into digital form.

For measuring temperature this sensor uses a Negative Temperature coefficient thermistor, which causes a decrease in its resistance value with increase in temperature. To get larger resistance value even for the smallest change in temperature, this sensor is usually made up of semiconductor ceramics or polymers.

The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy. The sampling rate of this sensor is 1Hz .i.e. it gives one reading for every second. DHT11 is small in size with operating voltage from 3 to 5 volts. The maximum current used while measuring is 2.5mA.

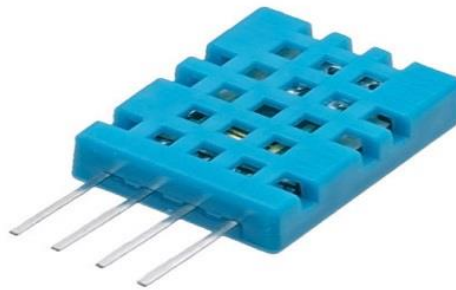


Fig 4.2 DHT11 Sensor

DHT11 sensor has four pins- VCC, GND, Data Pin and a not connected pin. A pull-up resistor of 5k to 10k ohms is provided for communication between sensor and micro-controller.

4.1.2 Applications

This sensor is used in various applications such as measuring humidity and temperature values in heating, ventilation and air conditioning systems. Weather stations also use these sensors to predict weather conditions. The humidity sensor is used as a preventive measure in homes where people are affected by humidity. Offices, cars, museums, greenhouses and industries use this sensor for measuring humidity values and as a safety measure.

4.2 IOT MODULE-NODEMCU(ESP8266):



Fig 4.3 IOT ESP8266

The ESP8266 module enables microcontrollers to connect to 2.4 GHz Wi-Fi, using IEEE 802.11 bgn. It can be used with ESP-AT firmware to provide Wi-Fi connectivity to external host MCUs, or it can be used as a self-sufficient MCU by running an RTOS-based SDK. The module has a full TCP/IP stack and provides the ability for data processing, reads and controls of GPIOs.

4.2.1 ESP8266 Specifications

This is based on the ESP-12 module, which we discuss below.

ESP8266	DESCRIPTION
---------	-------------

Core	1
------	---

Architecture	32 bits
--------------	---------

Clock	Xtensa LX106 80-160MHz
-------	------------------------

WiFi	IEEE802.11 b/g/n support for WPA and WPA2
------	---

Bluetooth	No
-----------	----

ESP8266 DESCRIPTION

RAM	160KB - 64KB Instruction - 96KB Data
Flash	Extern QSPI - 512KB A 4MB
GPIO	16
DAC	0
ADC	1

4.2.2 ESP8266 Functions

ESP8266 has many applications when it comes to the IoT. Here are just some of the functions the chip is used for:

- **Networking:** The module's Wi-Fi antenna enables embedded devices to connect to routers and transmit data
- **Data Processing:** Includes processing basic inputs from analog and digital sensors for far more complex calculations with an RTOS or Non-OS SDK
- **P2P Connectivity:** Create direct communication between ESPs and other devices using IoT P2P connectivity
- **Web Server:** Access pages written in HTML or development languages.

4.2.3 ESP8266 Applications

The ESP8266 modules are commonly found in the following IoT devices:

- Smart security devices, including surveillance cameras and smart locks
- Smart energy devices, including HVACs and thermostats
- Smart industrial devices, including Programmable Logic Controllers (PLCs)
- Smart medical devices, including wearable health monitors

Chip versus Modules versus Development Boards

As discussed above, the ESP8266 is just the name of the chip. There are essentially three formats you can buy this in:

- **ESP8266 Chip:** This is the basic chip manufactured by Espressif, which comes unshielded and needs to be soldered onto a module. This is unsuitable for most users, apart from perhaps volume device manufacturers that can factor this into the production process under the unit cost of a module.
- **ESP8266 Modules:** These are the surface-mountable modules that contain the chip, which are ready to be mounted onto an MCU, produced by Espressif, Ai-Thinker and certain other manufacturers. They are usually shielded and pre-approved by the FCC for use. This means they're a good option for device manufacturers looking to scale production.
- **ESP8266 Development Boards:** These are the complete IoT MCU development boards that have the modules preinstalled. They're used for developers and manufacturers to create prototypes during the design stage, before they start production. Development boards are produced by several different manufacturers and the specifications differ between models. Some core specifications to be aware of when assessing ESP8266 IoT development board options include:
 - GPIO pins
 - ADC pins
 - Wi-Fi antennas
 - LEDs
 - Shielding*
 - Flash Memory

4.3 LCD:

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations,

and there are no limitations for displaying custom characters, special and even animations, etc.



Fig 4.4 LCD 16*2

The 16×2 LCD pinout is shown below.

- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.
- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(0 = data mode, and 1 = command mode).
- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.
- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire

mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.

- Pin15 (+ve pin of the LED): This pin is connected to +5V
- Pin 16 (-ve pin of the LED): This pin is connected to GND.

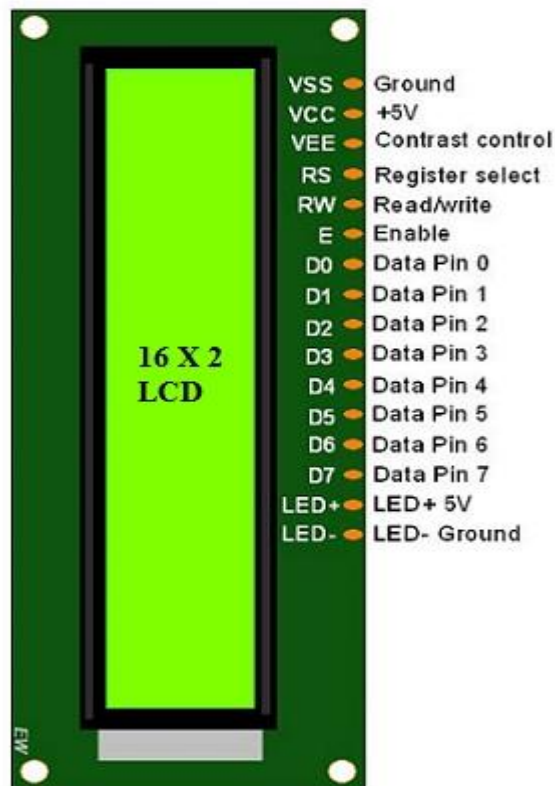


Fig 4.5 LCD-16×2-pin-diagram

Features of LCD16x2

The features of this LCD mainly include the following.

- The operating voltage of this LCD is 4.7V-5.3V
- It includes two rows where each row can produce 16-characters.
- The utilization of current is 1mA with no backlight
- Every character can be built with a 5×8 pixel box
- The alphanumeric LCDs alphabets & numbers
- Is display can work on two modes like 4-bit & 8-bit
- These are obtainable in Blue & Green Backlight

- It displays a few custom generated characters

Registers of LCD

A 16×2 LCD has two registers like data register and command register. The RS (register select) is mainly used to change from one register to another. When the register set is '0', then it is known as command register. Similarly, when the register set is '1', then it is known as data register.

4.3.1 Command Register

The main function of the command register is to store the instructions of command which are given to the display. So that predefined tasks can be performed such as clearing the display, initializing, set the cursor place, and display control. Here commands processing can occur within the register.

4.3.2 Data Register

The main function of the data register is to store the information which is to be exhibited on the LCD screen. Here, the ASCII value of the character is the information which is to be exhibited on the screen of LCD. Whenever we send the information to LCD, it transmits to the data register, and then the process will be starting there. When register set =1, then the data register will be selected.

4.4 SOLAR PANEL:

Solar PV Panels: A photovoltaic (PV) panel harnesses the solar radiation into electrical energy to be supplied to the electric vehicle batteries. In fast EV charging stations, large array (~300) of electrically connected solar panels would be installed to generate the required power to charge electric cars in a day. The angle of solar panel is one of the crucial factors in deciding the output efficiency of these panels. The other factors being the latitude, longitude and weather conditions. The solar panels are attached to moveable, motor-controlled solar trackers. Based on the inputs from the microcontrollers, these motor-run trackers move the solar panels axis to receive maximum sunlight any given time in a day.

4.4.1 PANEL RATINGS:

Single Panel rating: 20W 18V 1.12A

Total number of Panels = 6

Panel Connection:

3 Parallel ($3 \times 1.12 = 3.36\text{A}$)

1 Series ($18 + 18 = 36\text{V}$)

Panel Output = 120W 36V 3.36A

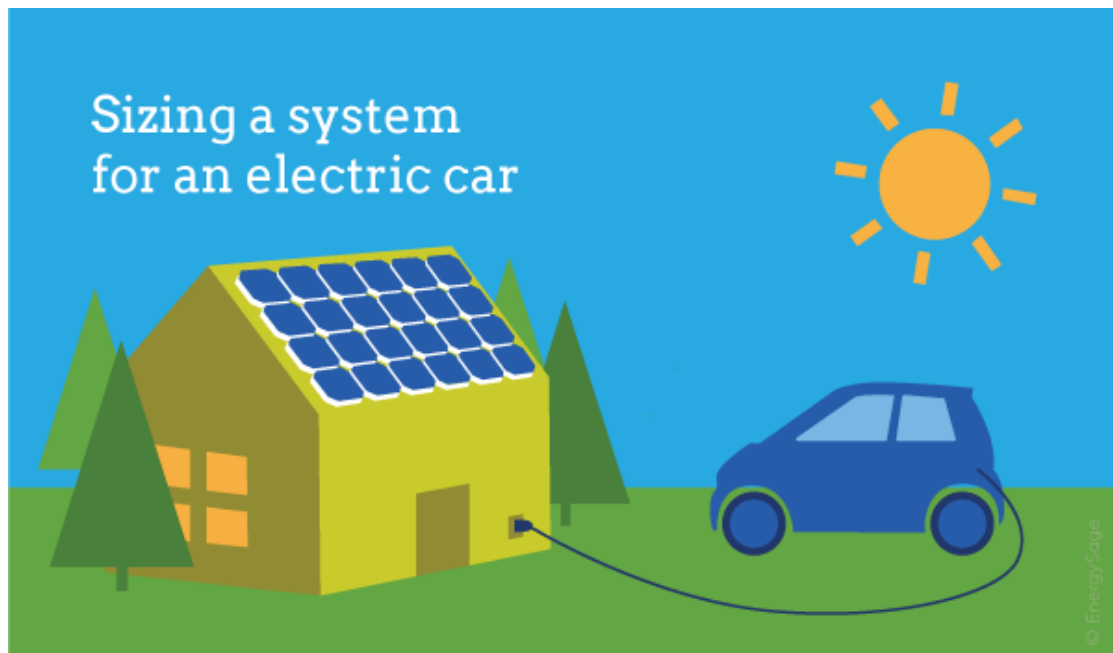


FIG 4.6 Panel designing model

4.4.2 Working Principle

Needless to say that the Sun is the biggest source of renewable energy for the Earth. The fact is that even though the earth receives only a part of the energy generated by the Sun (i.e. Solar energy), that part of solar energy is also tremendously huge. The Earth receives solar energy in the form of light and heat. But in today's world, the words 'power' and 'energy' are leaned more towards 'electricity'. This article explains how electricity is harvested from the solar energy and how it is utilized.

4.4.3 How Solar Power Works

Electrical energy can be harvested from solar power by means of either photovoltaics or concentrated solar power systems.

4.4.4 Photovoltaics (PV)

Photovoltaics directly convert solar energy into electricity. They work on the principle of the photovoltaic effect. When certain materials are exposed to light, they absorb photons and release free electrons. This phenomenon is called as the photoelectric effect. Photovoltaic effect is a method of producing direct current electricity based on the principle of the photoelectric effect.

Based on the principle of photovoltaic effect, solar cells or photovoltaic cells are made. They convert sunlight into direct current (DC) electricity. But, a single photovoltaic cell does not produce enough amount of electricity. Therefore, a number of photovoltaic cells are mounted on a supporting frame and are electrically connected to each other to form a photovoltaic module or solar panel. Commonly available solar panels range from several hundred watts (say 100 watts) up to few kilowatts (ever heard of a 5kW solar panel?). They are available in different sizes and different price ranges. Solar panels or modules are designed to supply electric power at a certain voltage (say 12v), but the current they produce is directly dependent on the incident light. As of now it is clear that photovoltaic modules produce DC electricity. But, for most of the times we require AC power and, hence, solar power system consists of an inverter too.

4.4.5 Photovoltaic Solar Power System

According to the requirement of power, multiple photovoltaic modules are electrically connected together to form a PV array and to achieve more power. There are different types of PV systems according to their implementation.

PV direct systems: These systems supply the load only when the Sun is shining. There is no storage of power generated and, hence, batteries are absent. An inverter may or may not be used depending on the type of load.

Off-grid systems: This type of system is commonly used at locations where power from the grid is not available or not reliable. An off-grid solar power system is not connected to any electric grid. It consists solar panel arrays, storage batteries and inverter circuits.

Grid connected systems: These solar power systems are tied with grids so that the excess required power can be accessed from the grid. They may or may not be backed by batteries.

4.5 SEPIC CONVERTER USING MPPT ALGORITHM

4.5.1 SEPIC CONVERTER

RATING:

Input-24volts

Output-42volts

A SEPIC is a cascaded boost/buck-boost converter, with its input stage similar to that of a basic boost converter, and its output stage is similar to that of a basic buck-boost converter. Overall, a SEPIC function is similar to a buck-boost converter, but has the additional advantages of having its output voltage polarity non-inverted with respect to its input voltage, having a true shutdown mode – i.e. when switch S turns off, the converter's output voltage reduces to 0V, and having isolation between the input and output.

The capacitive isolation prevents unwanted current from flowing from the input to output. Switch S 's duty cycle controls the SEPIC's output voltage. Typically, the switch S is an electrically controlled switch, such as a power metal oxide semiconductor field effect transistor (MOSFET), power bipolar junction transistor (BJT) or insulated-gate bipolar transistor (IGBT). Its switching actions are controlled by a pulse-width modulation (PWM) or pulse-frequency modulation (PFM) controller. A PWM controller varies switch S 's duty cycle while keeping its switching frequency constant; while a PFM controller varies S 's switching frequency while keeping its duty cycle constant.

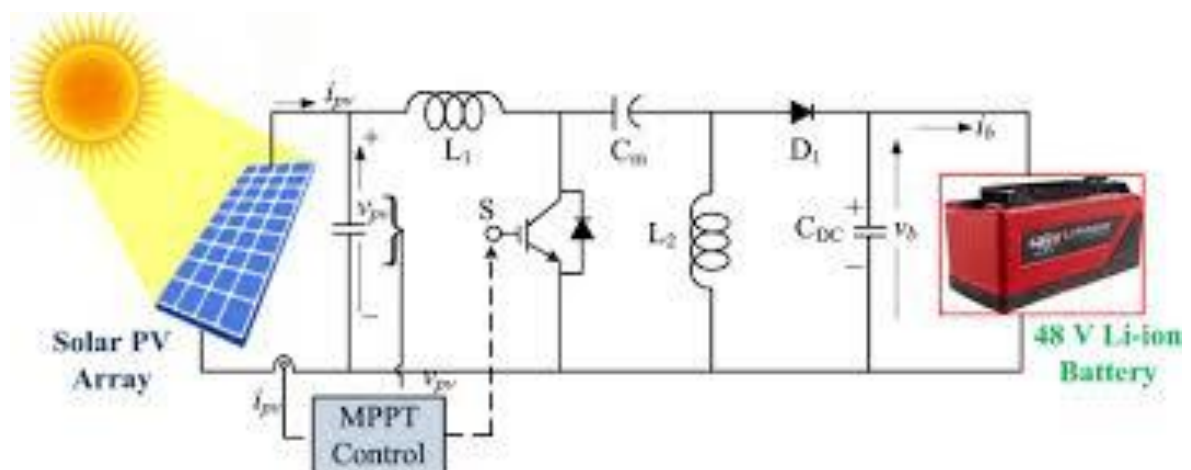


Fig 4.7 Working diagram of PV system with SEPIC converter

4.5.2 BASIC SEPIC OPERATION

The basic SEPIC performs DC-DC voltage conversion through energy exchange between its coupling capacitor and switching inductors (C_{in} , L_1 and L_2). The switch controls the energy exchange amount between the capacitor and inductors. Maximizing energy exchange efficiency and overall converter efficiency requires this SEPIC design operating in continuous conduction mode (CCM).

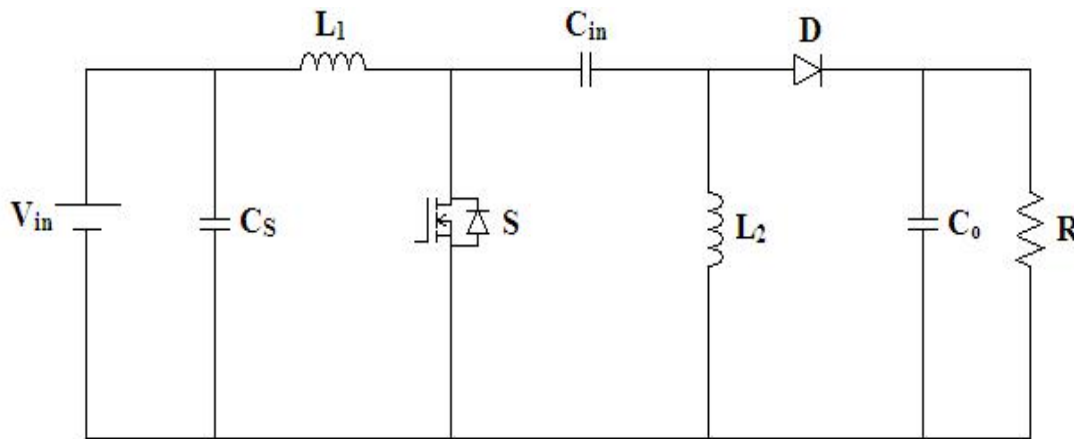


Fig 4.8 SEPI Converter

Figure 4.8 shows the circuit diagram of a SEPIC converter. There are two modes of operation in SEPIC. They are

1. Continuous Conduction Mode
2. Discontinuous Conduction Mode

4.5.3 Continuous Conduction Mode

In continuous conduction mode of operation, the inductor current will be continuous and never reaches to zero. That means operating the SEPIC in CCM means never letting the currents through L_1 and L_2 reduce to 0A – i.e. never letting L_1 and L_2 completely discharge. When the SEPI Reaches steady-state operation, the average voltage across C_{in} will be equal to that of V_{in} . Additionally, the average current through C_{in} is 0A in steady-state. When this steady-state phenomenon occurs, L_2 is the only source of current to the output load. Thus, L_2 's average current equals that of the output load, and is independent of V_{in} .

In CCM, the sum of the average voltages across the SEPIC's energy storage elements (excluding input and output filter capacitors C_s and C_o) equal that of the SEPIC's input voltage described as follows,

$$V_{in} = V_{L1} + V_{Cin} + V_{L2} \quad (4.1)$$

Since the average voltage across C_{in} equals that of V_{in} , V_{Cin} equals V_{in} , leading to

$$V_{L1} = V_{L2} \quad (4.2)$$

Under CCM in steady-state, the SEPIC's operation further splits into two operation modes: when switch S conducts and when it does not conduct. Analysing the SEPIC's entire operation in CCM requires analysing it in switch S 's conduction and non-conduction modes.

When switch is close

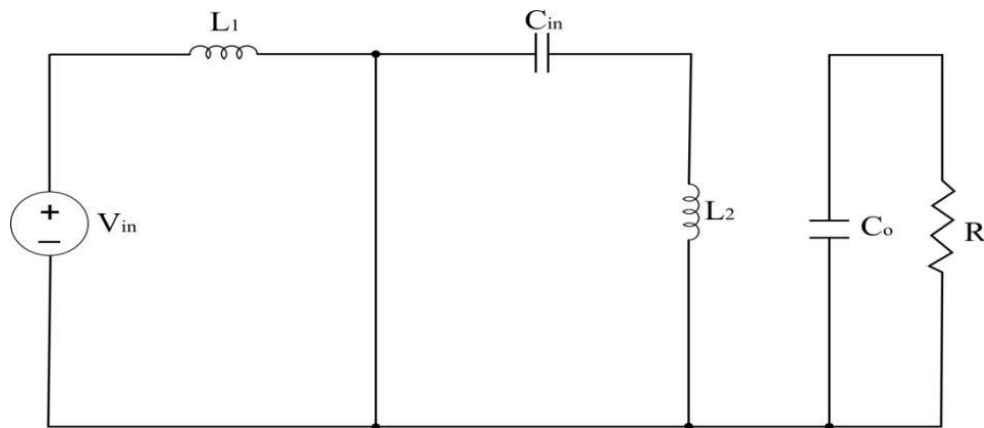


Fig 4.9 Switch on condition

Figure 4.9 shows the SEPIC's operation when switch S conducts. Filter capacitors C_s and C_o are assumed to be in steady-state, thus no current flows through these two components until they discharge. Furthermore, C_s and C_o are also assumed to be large enough in capacitance such that the SEPIC's input and output ripple voltages nearly comes to zero.

When switch S conducts during the first half-switching cycle, the current through $L1$ increases in the positive direction while the current through $L2$ increases in the negative direction. Hence $L1$ charges via V_{in} , while $L2$ discharges (acting as a source) through C_{in} . S remains closed for a short time period and during this time period the instantaneous voltage across C_{in} equals V_{in} . Thus, V_{L1} and V_{L2} both equal approximately V_{in} in magnitude.

The only difference between the two voltages is that V_{L2} 's polarity is reversed (i.e. negative) because L_2 is discharging. C_{in} in turn discharges and supplies current to L_2 in order to store energy in it, so that L_2 can supply current to the output load during the next half-switching cycle when switch S no longer conducts. During this entire half-switching cycle, diode D does not conduct i.e. in other words, it is reverse-biased. Thus C_o discharges and hence is the only component that helps maintain the output load current when switch S is conducting. During the second half-switching cycle, switch S turns off.

When switch is opened: Figure 4.10 shows the SEPIC operation when the switch is opened. At the end of one half-switching cycle, switch S turns off. The new path for the input current is through L_1 and C_{in} . Because current cannot change instantaneously through an inductor, Inductor Currents do not immediately change. Thus, capacitor current C_{in} equals L_1 current. L_2 continues to discharge, but during this half-switching cycle it discharges into C_o , thus turning on D i.e., forward biased and supplying current to the output load.

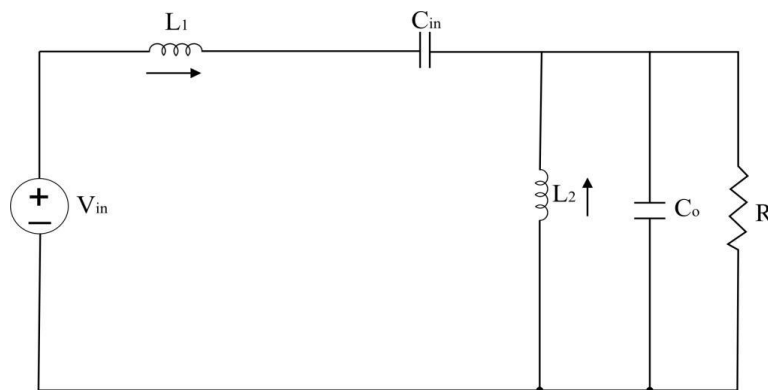


Fig 4.10 Switch off condition

However, the direction of L_2 current causes it to add to the input current that already flows to the output load. Thus, when S does not conduct, both L_1 and L_2 supply current to the output load. V_{in} and L_1 charge C_{in} (which discharged during the half-switching cycle when switch S conducted), and L_2 continues discharging to the output load until switch S conducts again at the beginning of the next half-switching cycle (when C_{in} supplies current to charge L_2).

4.5.4 Discontinuous Conduction Mode

In discontinuous conduction mode of operation, the inductor current reaches to zero. The discontinuous conduction mode occurs if the currents through L1 and L2 remain at 0A for any significant time period relative to switch switching period. Operating a SEPIC in DCM yields higher efficiency at lighter current loads, but requires operating at high current loads. Therefore, operating in CCM should yield better overall converter efficiency.

4.5.5 DESIGN OF SEPIC CONVERTER

Designing the SEPIC for proper operation under the proposed specifications requires selecting the proper component values.

Components in the SEPIC are inductors L1 and L2, switching transistor S, diode D and capacitor Cin. The input and output filter capacitors, Cs and Co, also need to be large enough in capacitance to minimize input and output voltage ripple. One major disadvantage with the SEPIC topology is that its output voltage ripple is inherently large because a pulsating diode (D) connects to its output. As mentioned, D conducts when switch S turns off and does not conduct when switch S turns on, resulting in a pulsating current fed to the converter's output. Hence Co needs to be large in capacitance to effectively quell any output voltage ripple that results from diode pulsating current.

Regarding switch, a MOSFET is preferable over a BJT because MOSFETs offer higher input impedance and lower voltage drop across its main current path compared to BJTs. Furthermore, a BJT needs to be biased with additional resistors as current differences control BJT switching, unlike MOSFETs, in which voltage differences control switching.

The formulas used in the design of the SEPIC converter are as follows,

INDUCTOR CURRENT

$$I_L = \frac{I_{OUT} \times V_o \times 40\%}{V_{IN(min)}} \quad (4.3)$$

INDUCTOR L1 & L2

$$L_1 = L_2 = \frac{V_{IN(min)} \times D_{max}}{\Delta I_L \times F_{SW}} \quad (4.4)$$

OUTPUT RIPPLE VOLTAGE

$$\Delta V_{\text{CIN}} = \frac{I_{\text{OUT(max)}}}{C_{\text{IN}} \times F_{\text{SW}}} \times \frac{V_{\text{OUT}}}{V_{\text{IN}} + V_{\text{OUT}} + V_{\text{D}}} \quad (4.5)$$

OUTPUT CAPACITOR

$$C_{\text{OUT}} = \frac{I_{\text{OUT}} \times D_{\text{max}}}{V_{\text{RIP}} \times 0.5 \times F_{\text{SW}}} \quad (4.6)$$

Based on the above equations, the design parameters are calculated which helps to control the input of commutation circuit in Brushless DC motor.

4.5.6 MAXIMUM POWER POINT TRACKING

Maximum power point tracking (MPPT) is an algorithm implemented in photovoltaic (PV) inverters to continuously adjust the impedance seen by the solar array to keep the PV system operating at, or close to, the peak power point of the PV panel under varying conditions, like changing solar irradiance, temperature, and load.

Engineers developing solar inverters implement MPPT algorithms to maximize the power generated by PV systems. The algorithms control the voltage to ensure that the system operates at “maximum power point” (or peak voltage) on the power voltage curve, as shown below.

MPPT algorithms are typically used in the controller designs for PV systems. The algorithms account for factors such as variable irradiance (sunlight) and temperature to ensure that the PV system generates maximum power at all times.

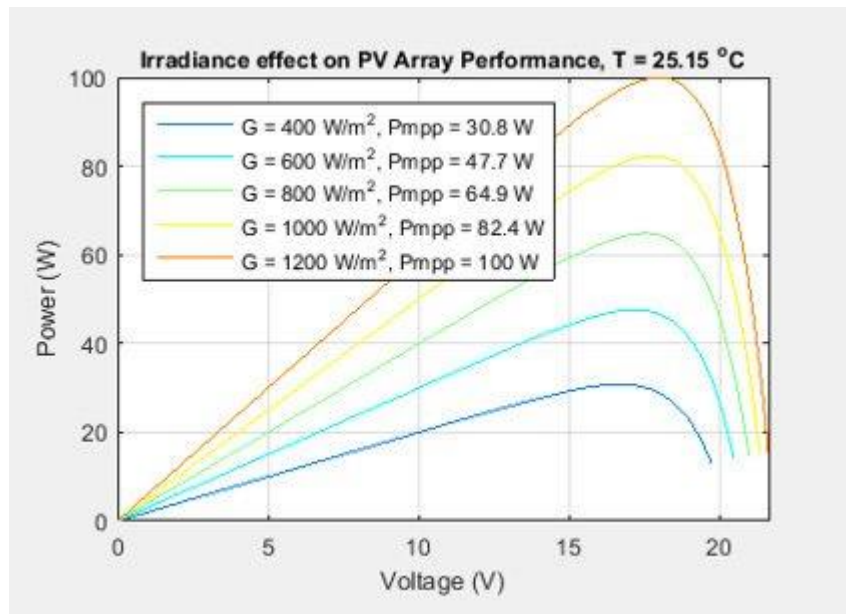


Fig 4.11 Power/voltage curve

Power voltage curve with I-V and P-V characteristics of a photovoltaic system.

The three most common MPPT algorithms are:

1. **Perturbation and observation (P&O):** This algorithm perturbs the operating voltage to ensure maximum power. While there are several advanced and more optimized variants of this algorithm, a basic P&O MPPT algorithm is shown below.

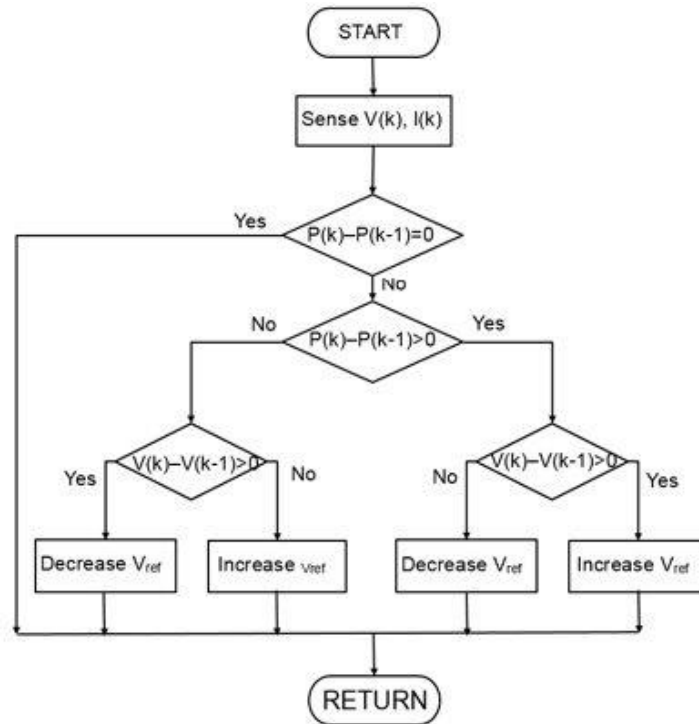


Fig 4.12 Basic P&O algorithm flow chart

4.6 AC-DC Converter Using Bridgeless SEPIC

RATING:

Input AC=24volts

Output dc=42volts

A new single phase PFC bridgeless rectifier is operated in discontinuous conduction mode. The DCM operation results in soft turn – on switching and relatively low inrush current. The voltage gain can be extended without extreme duty cycle. The proposed bridgeless rectifier is coupled magnetic configurations, results in higher overall efficiency and higher power density. The bridgeless configuration will reduce the conduction losses and the multiplier cell (D1, C3) and (D2 , C3) will increase the gain and reduce the switch voltage stress. The proposed circuit consist of two MOSFET switches (Q1, Q2) and two slow diodes (Dp ,Dn).

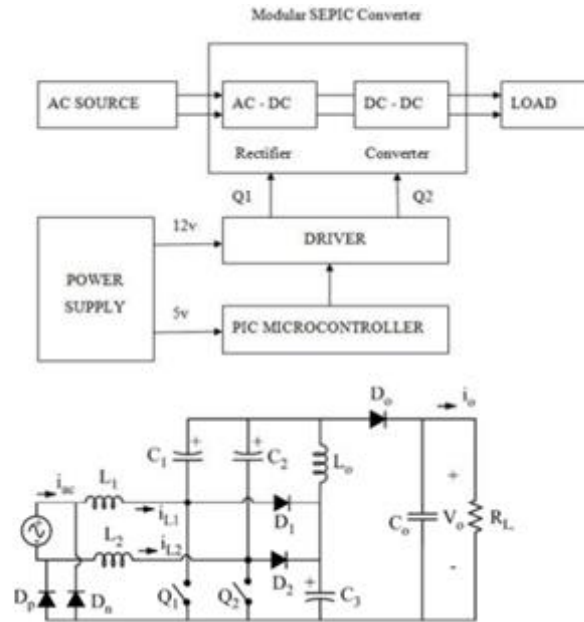


Fig 4.13 Bridgeless rectifier

The advantages of this system having simple circuitry, lower voltage stress in whole operation, conduction losses is reduced and higher efficiency.

4.6.1 MODE OF OPERATION

Bridgeless SEPIC PFC rectifier is having two switches Q1, Q2 and two diodes are serially connected with inductor in every half cycle of the operation. To show in the positive and negative half line periods equivalent circuit diagram.

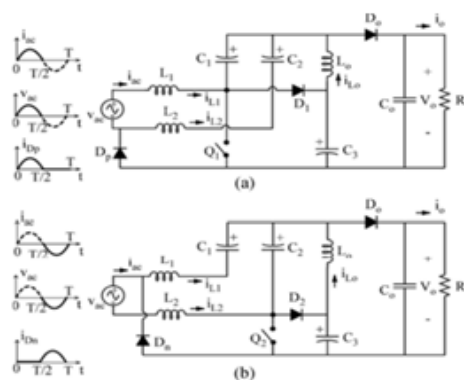


Fig 4.14 Modes of operation of rectifier

POSITIVE HALF –LINE MODE: Operation Since the proposed circuit consist of two symmetrical configurations as illustrated in Fig.4, the circuit is analysed for the positive half line cycle configuration shown infig.4a Assuming that the three inductors

are operating in DCM, then the circuit operation during one switching period T_s in a positive half – line period can be divided into three distinct operating modes as shown in Fig 5 (a) – (c), and it can be described as follows

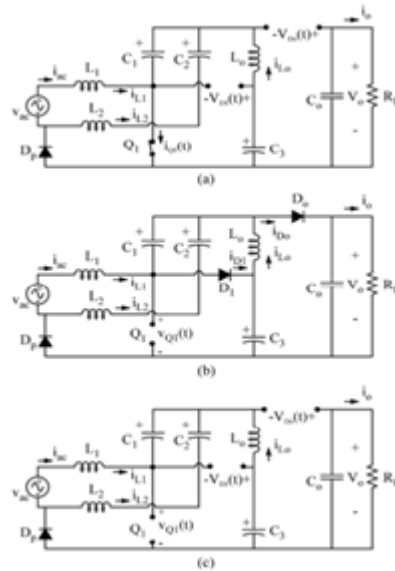


Fig 4.15 Topologies of rectifier

TABLE 1 Classification of stages according to switch, diodes, inductor current

No of stages	Switches	Diodes	Inductor current
First stage	Q_1, T_{on}	D_1, D_0 is reverse biased	$\frac{di_{Ln}}{dt} = \frac{V_{ac}}{L_n}$
Second stage	Q_2, T_{off}	D_1, D_0 is forward biased	$\frac{di_{Ln}}{dt} = -\frac{L_c}{L_n}$
Third stage	Q_2, T_{off}	D_1, D_2 reverse biased	constant

First Stage: In this stage, switch Q_1 is turned-on by the control signal and both diodes D_1 and D_0 are reversed biased as shown in Fig. 4(a). In this stage, the three-inductor currents increase linearly at a rate proportional to the input voltage

Second Stage: During this subinterval, switch Q1 is turned-off and both diodes D1 and Do will conduct simultaneously providing a path for the three inductors' currents as shown in Fig. 4(b). In this stage, the three inductors' currents decrease linearly at a rate proportional to the capacitor C1 voltage VC1. This stage ends when the sum of the currents flowing in the inductors addsup to zero, hence diodes D1 and Do are reverse biased $\frac{di_L}{dt} = -\frac{v_{C1}}{L_n}$, $n = 1, 2, o$. (2)

Third Stage: In this stage, switch Q1 remains turned-off while both diodesD1 and Do are reverse biased as illustrated in Fig. 4(c). Diode Dp provides a path for iLo. The three inductors behave as current sources, which keeps the currents constant. Hence, the voltage across the three inductors is zero. This period ends when switchQ1 is turned-on initiating the next turn-on of the switching cycle. Fig. 5 illustrates the theoretical DCM waveforms during one switching period Ts for the proposed rectifier.

4.7 PROTECTION CIRCUIT:

A **current sensor circuit** is a circuit that can measure the current flowing through it. Current sensor circuits are used extensively in systems such as battery management systems in order to detect the current to monitor for overcurrent, a short circuit, and the state of charge of the battery system. This keeps the system safe and can protect the system from devastating, dangerous conditions such as fires.

There's really 2 main approaches to building current sensor circuits for systems such as BMS.

The first approach is using a current sensor resistor, which is also called a shunt resistor or resistive shunt. This is a resistor of a small value (a few milliohms), highly precise, and of high power rating. Current sense resistors are used extensively in low power system that utilize less than 100A. It is low cost and relatively simple to implement.

The second approach is using a Hall Effect sensor. This is used more for high power systems, which utilize more than 100A such as electric vehicles. The Hall Effect sensor is more costly and complex to implement. An advantage to using the Hall Effect sensor method is its ability to work with higher current and it provides electrical isolation if a particular device has multiple sources of power. We will focus on building a current sensor circuit using a current sense resistor. So current sense resistors are widely available online and most cost less than \$2, with the majority being under \$1.

When we are designing a current sensor circuit that will be based off of a current sense resistor, there are a number of parameters that need to be considered. It's best to examine the datasheet and specs of the current sense resistor before purchasing to make sure that it suits all your needs.

The first thing you know is what the maximum current that your system can reach. You have to know maximum current conditions because you have to make sure that the current sense resistor doesn't exceed a certain power threshold. Just like regular resistors, current sense resistors have power ratings that should never be exceeded.

So doing the math, $\text{power} = I^2R$.

So if we know the maximum current conditions and you multiply it by the value of the current sense resistor you are purchasing, the values should not exceed the power rating of that resistor. Or, in other words, the value of the power rating of the current sense resistor should be higher than the power you calculated for maximum current conditions.

Another thing that is important is the tolerance of the resistor. Most current sense resistors have with a tolerance of $\pm 1\%$. Highly specialized ones may come with an even lower tolerance. Depending on how precise you need your current sensing to be may alter your pick here but for most basic applications $\pm 1\%$ should suffice well.

The best current sense resistors will have a low value resistance value, a high power rating, and a low tolerance level. The lower the resistance value, the better,

because the less power it uses, compared to a higher value resistor, which translates to less power loss.

For this circuit, we will select the Vishay WSLP59312L000FEA current sense resistor.

This is a 2 m Ohm, 8W, 1% tolerance current sense resistor.

Let's say we are working with a battery management system that is designed for a maximum of 62.5A.

Doing the math, $\text{Power} = I^2 R = 7.8125W$

So this is good, because our current sense resistor has a 8W power rating, so this will suffice for our application. Again, make sure you know maximum current conditions and do the necessary calculations to ensure power ratings are not exceeded, so that you don't create a dangerous circuit.

So how a current sensor works is that it's put in series with the circuit.

You have your battery and then you can have the load and current sensor in series. So current is the same in series. So being that we have the current sense resistor connected in series to the battery management system, we know that is the same current that is output by the BMS. So from this configuration, we can always know what current the BMS is outputting.

An important thing to note is that the output from the current sensor circuit must be connected to a chip in order to read the voltage to be able to determine the current. A chip cannot directly measure current. Instead, it measure the voltage drop across the current sense resistor and using ohm's law can compute the current, according to the formula, $I = V/R$. However, it must be noted as well that the voltage drop across the current sensor is going to be very low, because the resistance value is very low. So the voltage has to be amplified first before it gets read by a chip.

If you have a specialty microcontroller chip, it may be able to do the amplification for you. In that case, you would simply have to create the current sensor circuit and connect its output to the chip without any external amplification

circuit, being that the chip has internal circuitry that can amplify it. If not, you have to create your own external circuit to do the amplification.

We examine both approaches below.

Direct To Microcontroller

So if your circuit doesn't need any external amplification and you can connect it directly to a microcontroller that will amplify the voltage across the current sense resistor, then the circuit shown below can be built.

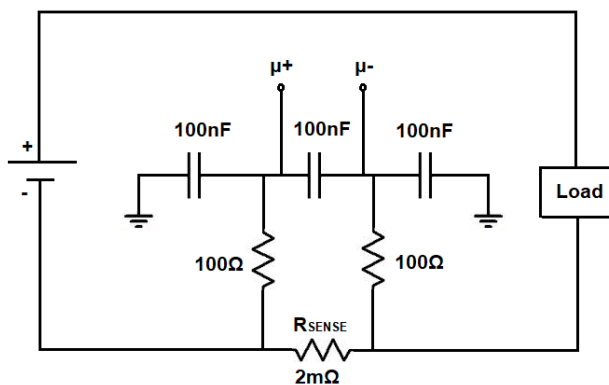


Fig 4.16 Filter circuit

Other than the current sense resistor found at the bottom, the other resistors and the capacitors are for filtering.

The $\mu+$ and $\mu-$ connections are to connect to the microcontroller which will then read the voltage from the current sensor circuit. Again, no amplification is done in this circuit. This is only if you are connecting to a chip that provides amplification in its internal circuitry.

4.7.1 External Amplification Circuit

Now in this circuit, you as the circuit builder provide the amplification from the output of the voltage drop across the current sense resistor. This amplification can be achieved through the use of a precise op amp. A good op amp chip that can be used is the MCP6022, which is a general purpose operational amplifier. It provides dual op amp use, but for this circuit, only one op amp is needed

The datasheet for the MCP6022 operational amplifier is shown at the following link: [MCP6022 Operational Amplifier Datasheet](#).

The pinout for the MCP6022 amplifier is shown below.

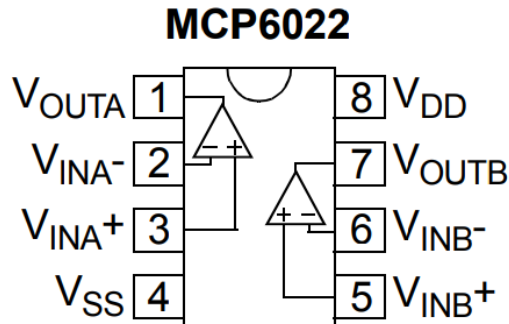


FIG 4.17 Pin diagram of OP-AMP

The op amp circuit we will build is shown below.

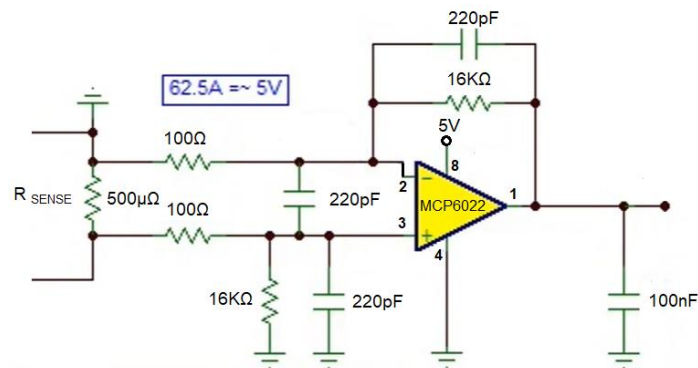


FIG 4.18 OP-AMP design

So this circuit is a rugged and proven circuit for current sense resistor circuits. Notice that the input resistor is 100Ω and the output resistor is 16KΩ; this gives us a gain of 161. The capacitors are for 220pF capacitors are for filtering. The 100nF capacitor at the output is also for filtering.

So now you connect the output of this op amp, which is pin 1 to the input of the IC that will read this voltage and controls the battery management system. If you desire a more in-depth explanation of this circuit, see Current Sensor Circuit with an External Amplifier.

So these are 2 circuits both utilizing current sense resistors in order to sense current, one that doesn't provide amplification because it will be connected to a chip

that will provide the amplification of the current sense resistor voltage drop and the other with an external op amp circuitry that provides amplification which can then be connected to a chip that doesn't need to do any amplification but only read and manage the system conditions.

4.8 MICROCONTROLLER

The microcontrollers used in a typical solar PV charging system have the responsibility of:

- Ensuring maximum efficiency of solar power generation through the MPPT controllers.
- Storing the extra energy generated from the solar PV panels into storage battery cells which.

Additionally, the microcontrollers are programmed to control the motor using a Pulse Width Modulation (PWM) technique to align the panels for maximum power generation.

4.8.1 DSPIC30F2010:

28-PIN DSPIC30F2010 ENHANCED FLASH 16-BIT DIGITAL SIGNAL CONTROLLER

PIN DIAGRAM:

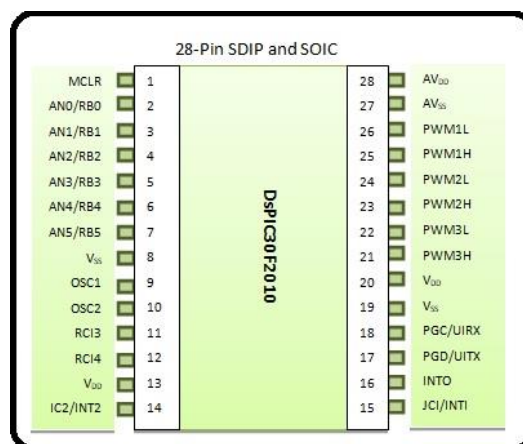


FIG 4.19 Pin diagram of microcontroller

4.8.2 HIGH-PERFORMANCE MODIFIED RISC CPU:

- Modified Harvard architecture

- C compiler optimized instruction set architecture
- 84 base instructions with flexible addressing modes
- 24-bit wide instructions, 16-bit wide data path
- 12 Kbytes on-chip Flash program space
- 512 bytes on-chip data RAM
- 1 Kbyte non-volatile data EEPROM
- 16 x 16-bit working register array
- Up to 30 MIPS operation:
 - DC to 40 MHz external clock input
 - 4 MHz-10 MHz oscillator input with PLL active (4x, 8x, 16x)
- 27 interrupt sources
- Three external interrupt sources
- 8 user selectable priority levels for each interrupt
- 4 processor exceptions and software traps

4.8.3 DSP ENGINE FEATURES:

- Modulo and Bit-Reversed modes
- Two, 40-bit wide accumulators with optional saturation logic
- 17-bit x 17-bit single cycle hardware fractional/ integer multiplier
- Single cycle Multiply-Accumulate (MAC) operation
- 40-stage Barrel Shifter
- Dual data fetch

4.8.4 MOTOR CONTROL PWM MODULE FEATURES:

- 6 PWM output channels
 - Complementary or Independent Output modes
 - Edge and Centre Aligned modes
- 4 duty cycle generators
- Dedicated time base with 4 modes
- Programmable output polarity

- Dead time control for Complementary mode
- Manual output control
- Trigger for synchronized A/D conversions

4.8.5 QUADRATURE ENCODER INTERFACE MODULE FEATURES:

- Phase A, Phase B and Index Pulse input
- 16-bit up/down position counter
- Count direction status
- Position Measurement (x2 and x4) mode
- Programmable digital noise filters on inputs
- Alternate 16-bit Timer/Counter mode
- Interrupt on position counter rollover/underflow
- Analog Features:
- 10-bit Analog-to-Digital Converter (A/D) with:
 - 500 Ksps (for 10-bit A/D) conversion rate
 - Six input channels
 - Conversion available during Sleep and Idle

4.8.6 PROGRAMMABLE BROWN-OUT DETECTION AND RESET GENERATION SPECIAL MICROCONTROLLER FEATURES:

- **ENHANCED FLASH PROGRAM MEMORY:**
 - 10,000 erase/write cycle (min.) for industrial temperature range, 100K (typical)
- **DATA EEPROM MEMORY:**
 - 100,000 erase/write cycle (min.) for industrial temperature range, 1M (typical)
- Self-reprogrammable under software control
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Flexible Watchdog Timer (WDT) with on-chip low power RC oscillator for reliable operation

- Fail-Safe clock monitor operation
- Detects clock failure and switches to on-chip low power RC oscillator
- Programmable code protection
- In-Circuit Serial Programming™ (ICSP™)
- Selectable Power Management modes - Sleep, Idle and Alternate Clock modes
- **CMOS Technology:**
 - Low power, high speed Flash technology
 - Wide operating voltage range (2.5V to 5.5V)
 - Industrial and Extended temperature ranges
 - Low power consumption

4.8.7 BLOCK DIAGRAM:

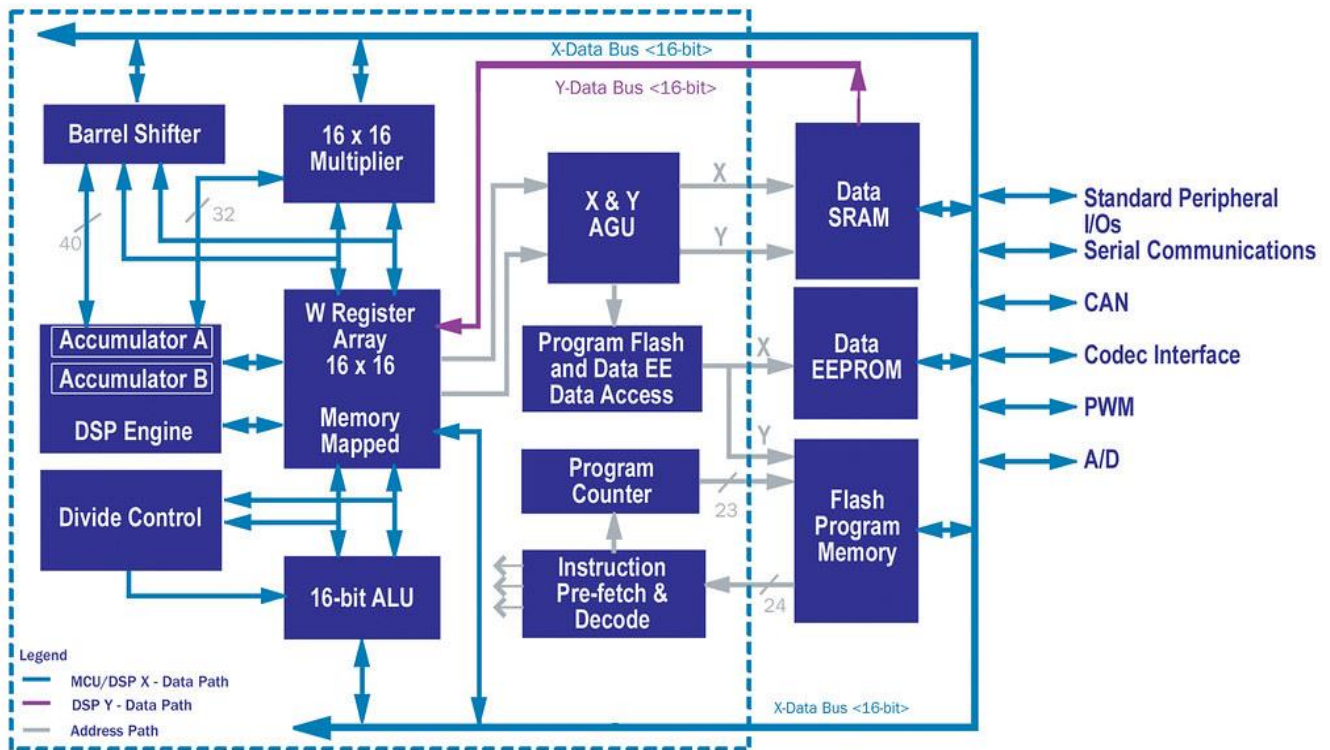


Fig 4.20 Block diagram of microcontroller

4.8.8 PIN DESCRIPTION:

PIN # (n)	PIN NAME + DESCRIPTION	Port
1	MCLR	Master Clear (Reset) input or programming voltage input.
2	EMUD3/AN0/VREF+/CN2/RB0	B
3	EMUC3/AN1/VREF-/CN3/RB1	B
4	AN2/SS1/CN4/RB2	B
5	AN3/INDX/CN6/RB3	B
6	AN4/QEA/IC7/CN6/RB4	B
7	AN5/QEB/IC8/CN7/RB5	B
8	Vss	Ground reference for analogue module
9	OSC1/CLKI	Oscillator Input
10	OSC2/CLKO/RC15	Oscillator
11	EMUD1/SOSCI/T2CK/U1ATX/CN1/RC13	C
12	EMUC1/SOSCO/T1CK/U1ARX/CN0/RC14	C
13	RC2/CCP1/VDD1A	C
14	EMUD2/OSC2/IC2/INT2/RD1	D
15	EMUC2/OC1/IC1/INT1/RD0	D

16	FLTA/INT0/SCK1/OCFA/RE8	E
17	PGD/EMUD/U1TX/SDO1/SCL/RF3	F
18	PGC/EMUC/U1RX/SDI1/SDA/RF2	F
19	V _{SS}	Ground
20	V _{DD}	Positive Power Supply for analogue module
21	PWM3H/RE5	E
22	PWM3L/RE4	E
23	PWM2H/RE3	E
24	PWM2L/RE2	E
25	PWM1H/RE1	E
26	PWM1L/RE0	E
27	AV _{SS}	Analogue Ground
28	AV _{DD}	Analogue Power Supply

4.9 CORE OVERVIEW

The core has a 24-bit instruction word. The Program Counter (PC) is 23 bits wide with the Least Significant (LS) bit always clear (see Section 3.1), and the Most Significant (MS) bit is ignored during normal program execution, except for certain specialized instructions. Thus, the PC can address up to 4M instruction words of user program space. An instruction pre-fetch mechanism is used to help maintain throughput. Program loop constructs, free from loop count management overhead, are supported using the DO and REPEAT instructions, both of which are

interruptible at any point. The working register array consists of 16x16-bit registers, each of which can act as data, address or offset registers. One working register (W15) operates as a software stack pointer for interrupts and calls.

The data space is 64 Kbytes (32K words) and is split into two blocks, referred to as X and Y data memory. Each block has its own independent Address Generation Unit (AGU). Most instructions operate solely through the X memory AGU, which provides the appearance of a single unified data space. The Multiply-Accumulate (MAC) class of dual source DSP instructions operate through both the X and Y AGUs, splitting the data address space into two parts (see Section 3.2). The X and Y data space boundary is device specific and cannot be altered by the user. Each data word consists of 2 bytes, and most instructions can address data either as words or bytes.

There are two methods of accessing data stored in program memory:

- The upper 32 Kbytes of data space memory can be mapped into the lower half (user space) of program space at any 16K program word boundary, defined by the 8-bit Program Space Visibility Page (PSVPAG) register. This lets any instruction access program space as if it were data space, with a limitation that the access requires an additional cycle. Moreover, only the lower 16 bits of each instruction word can be accessed using this method.
- Linear indirect access of 32K word pages within program space is also possible using any working register, via table read and write instructions. Table read and write instructions can be used to access all 24 bits of an instruction word. Overhead-free circular buffers (modulo addressing) are supported in both X and Y address spaces. This is primarily intended to remove the loop overhead for DSP algorithms.

The X AGU also supports bit-reversed addressing on destination effective addresses, to greatly simplify input or output data reordering for radix-2 FFT algorithms. Refer to Section 4.0 for details on modulo and bit-reversed addressing. The core supports Inherent (no operand), Relative, Literal, Memory Direct, Register Direct, Register Indirect, Register Offset and Literal Offset Addressing modes.

Instructions are associated with predefined Addressing modes, depending upon their functional requirements.

For most instructions, the core is capable of executing a data (or program data) memory read, a working register (data) read, a data memory write and a program (instruction) memory read per instruction cycle. As a result, 3-operand instructions are supported, allowing $C = A+B$ operations to be executed in a single cycle.

A DSP engine has been included to significantly enhance the core arithmetic capability and throughput. It features a high speed 17-bit by 17-bit multiplier, a 40-bit ALU, two 40-bit saturating accumulators and a 40-bit bidirectional barrel shifter. Data in the accumulator or any working register can be shifted up to 15 bits right or 16 bits left in a single cycle. The DSP instructions operate seamlessly with all other instructions and have been designed for optimal real-time performance. The MAC class of instructions can concurrently fetch two data operands from memory, while multiplying two W registers. To enable this concurrent fetching of data operands, the data space has been split for these instructions and linear for all others. This has been achieved in a transparent and flexible manner, by dedicating certain working registers to each address space for the MAC class of instructions.

The core does not support a multi-stage instruction pipeline. However, a single stage instruction pre-fetch mechanism is used, which accesses and partially decodes instructions a cycle ahead of execution, in order to maximize available execution time. Most instructions execute in a single cycle, with certain exceptions.

The core features a vectored exception processing structure for traps and interrupts, with 62 independent vectors. The exceptions consist of up to 8 traps (of which 4 are reserved) and 54 interrupts. Each interrupt is prioritized based on a user assigned priority between 1 and 7 (1 being the lowest priority and 7 being the highest) in conjunction with a predetermined 'natural order'. Traps have fixed priorities, ranging from 8 to 15.

4.9.1 PROGRAMMER'S MODEL

The programmer's model is shown in Figure 2-1 and consists of 16x16-bit working registers (W0 through W15), 2x40-bit accumulators (AccA and AccB), STATUS register (SR), Data Table Page register (TBLPAG), Program Space

Visibility Page register (PSVPAG), DO and REPEAT registers (DOSTART, DOEND, DCOUNT and RCOUNT), and Program Counter (PC). The working registers can act as data, address or offset registers. All registers are memory mapped. W0 acts as the W register for file register addressing.

Some of these registers have a shadow register associated with each of them, as shown in Figure 2-1. The shadow register is used as a temporary holding register and can transfer its contents to or from its host register upon the occurrence of an event. None of the shadow registers are accessible directly. The following rules apply for transfer of registers into and out of shadows.

- PUSH.S and POP.S

W0, W1, W2, W3, SR (DC, N, OV, Z and C bits only) are transferred.

- DO instruction

DOSTART, DOEND, DCOUNT shadows are pushed on loop start, and popped on loop end.

When a byte operation is performed on a working register, only the Least Significant Byte of the target register is affected. However, a benefit of memory mapped working registers is that both the Least and Most Significant Bytes can be manipulated through byte wide data memory space accesses.

4.9.2 SOFTWARE STACK POINTER/ FRAME POINTER

The dsPIC devices contain a software stack. W15 is the dedicated software stack pointer (SP), and will be automatically modified by exception processing and subroutine calls and returns. However, W15 can be referenced by any instruction in the same manner as all other W registers. This simplifies the reading, writing and manipulation of the stack pointer (e.g., creating stack frames).

Note: In order to protect against misaligned stack accesses, W15<0> is always clear.

W15 is initialized to 0x0800 during a Reset. The user may reprogram the SP during initialization to any location within data space.

W14 has been dedicated as a stack frame pointer as defined by the LNK and ULNK instructions. However, W14 can be referenced by any instruction in the same manner as all other W registers.

STATUS REGISTER

The dsPIC core has a 16-bit status register (SR), the LS Byte of which is referred to as the SR Low Byte (SRL) and the MS Byte as the SR High Byte (SRH). See Figure 2-1 for SR layout.

SRL contains all the MCU ALU operation status flags (including the Z bit), as well as the CPU Interrupt Priority Level status bits, $IPL<2:0>$, and the REPEAT active status bit, RA. During exception processing, SRL is concatenated with the MS Byte of the PC to form a complete word value which is then stacked. The upper byte of the STATUS register contains the DSP Adder/Subtractor status bits, the DO Loop Active bit (DA) and the Digit Carry (DC) status bit.

4.9.3 PROGRAM COUNTER

The Program Counter is 23 bits wide. Bit 0 is always clear. Therefore, the PC can address up to 4M instruction words.

Divide Support

The dsPIC devices feature a 16/16-bit signed fractional divide operation, as well as 32/16-bit and 16/16-bit signed and unsigned integer divide operations, in the form of single instruction iterative divides. The following instructions and data sizes are supported:

1. DIVF – 16/16 signed fractional divide
2. DIV.sd – 32/16 signed divide
3. DIV.ud – 32/16 unsigned divide
4. DIV.sw – 16/16 signed divide
5. DIV.uw – 16/16 unsigned divide

The 16/16 divides are similar to the 32/16 (same number of iterations), but the dividend is either zero-extended or sign-extended during the first iteration.

The divide instructions must be executed within a REPEAT loop. Any other form of execution (e.g. a series of discrete divide instructions) will not function correctly because the instruction flow depends on RCOUNT. The divide instruction does not automatically set up the RCOUNT value, and it must, therefore, be explicitly and correctly specified in the REPEAT instruction, as shown in Table 2-1 (REPEAT will execute the target instruction {operand value+1} times). The REPEAT loop count must be set up for 18 iterations of the DIV/DIVF instruction. Thus, a complete divide operation requires 19 cycles.

4.9.4 MULTIPLIER

The 17x17-bit multiplier is capable of signed or unsigned operation and can multiplex its output using a scaler to support either 1.31 fractional (Q31) or 32-bit integer results. Unsigned operands are zero-extended into the 17th bit of the multiplier input value. Signed operands are sign-extended into the 17th bit of the multiplier input value. The output of the 17x17-bit multiplier/ scaler is a 33-bit value, which is sign-extended to 40 bits. Integer data is inherently represented as a signed two's complement value, where the MSB is defined as a sign bit. Generally speaking, the range of an N-bit two's complement integer is -2^{N-1} to $2^{N-1} - 1$. For a 16-bit integer, the data range is -32768 (0x8000) to 32767 (0x7FFF), including 0. For a 32-bit integer, the data range is -2,147,483,648 (0x8000 0000) to 2,147,483,645 (0x7FFF FFFF).

When the multiplier is configured for fractional multiplication, the data is represented as a two's complement fraction, where the MSB is defined as a sign bit and the radix point is implied to lie just after the sign bit (QX for-mat). The range of an N-bit two's complement fraction with this implied radix point is -1.0 to $(1-2^{1-N})$. For a 16-bit fraction, the Q15 data range is -1.0 (0x8000) to 0.999969482 (0x7FFF), including 0 and has a precision of 3.01518×10^{-5} . In Fractional mode, a 16x16 multiply operation generates a 1.31 product, which has a precision of 4.65661×10^{-10} . The same multiplier is used to support the MCU multi-ply instructions, which include integer 16-bit signed, unsigned and mixed sign multiplies.

The MUL instruction may be directed to use byte or word sized operands. Byte operands will direct a 16-bit result, and word operands will direct a 32-bit result to the specified register(s) in the W array.

4.9.5 DATA ACCUMULATORS AND ADDER/SUBTRACTOR

The data accumulator consists of a 40-bit adder/ subtractor with automatic sign extension logic. It can select one of two accumulators (A or B) as its pre-accumulation source and post-accumulation destination. For the ADD and LAC instructions, the data to be accumulated or loaded can be optionally scaled via the barrel shifter, prior to accumulation.

4.9.6 Adder/Subtractor, Overflow and Saturation

The adder/subtractor is a 40-bit adder with an optional zero input into one side and either true or complement data into the other input. In the case of addition, the carry/borrow input is active high and the other input is true data (not complemented), whereas in the case of subtraction, the carry/borrow input is active low and the other input is complemented. The adder/subtractor generates overflow status bits SA/SB and OA/OB, which are latched and reflected in the status register.

- Overflow from bit 39: this is a catastrophic overflow in which the sign of the accumulator is destroyed.
- Overflow into guard bits 32 through 39: this is a recoverable overflow. This bit is set whenever all the guard bits are not identical to each other.

The adder has an additional saturation block which controls accumulator data saturation, if selected. It uses the result of the adder, the overflow status bits described above, and the SATA/B (CORCON<7:6>) and ACCSAT (CORCON<4>) mode control bits to determine when and to what value to saturate. Six status register bits have been provided to support saturation and overflow; they are:

1. OA: AccA overflowed into guard bits
2. OB: AccB overflowed into guard bits
3. SA: AccA saturated (bit 31 overflow and saturation) or
AccA overflowed into guard bits and saturated (bit 39 overflow and saturation)
4. SB: AccB saturated (bit 31 overflow and saturation) or

AccB overflowed into guard bits and saturated (bit 39 overflow and saturation)

5. OAB: Logical OR of OA and OB

6. SAB: Logical OR of SA and SB

The OA and OB bits are modified each time data passes through the adder/Subtractor. When set, they indicate that the most recent operation has overflowed into the accumulator guard bits (bits 32 through 39). The OA and OB bits can also optionally generate an arithmetic warning trap when set and the corresponding overflow trap flag enable bit (OVATEN, OVBTEN) in the INTCON1 register (refer to Section 5.0) is set. This allows the user to take immediate action, for example, to correct system gain.

The SA and SB bits are modified each time data passes through the adder/subtractor, but can only be cleared by the user. When set, they indicate that the accumulator has overflowed its maximum range (bit 31 for 32-bit saturation, or bit 39 for 40-bit saturation) and will be saturated (if saturation is enabled). When saturation is not enabled, SA and SB default to bit 39 overflow and thus indicate that a catastrophic overflow has occurred. If the COVTE bit in the INTCON1 register is set, SA and SB bits will generate an arithmetic warning trap when saturation is disabled.

The overflow and saturation status bits can optionally be viewed in the Status Register (SR) as the logical OR of OA and OB (in bit OAB) and the logical OR of SA and SB (in bit SAB). This allows programmers to check one bit in the Status Register to determine if either accumulator has overflowed, or one bit to determine if either accumulator has saturated. This would be useful for complex number arithmetic which typically uses both the accumulators.

The device supports three Saturation and Overflow modes.

1. Bit 39 Overflow and Saturation:

When bit 39 overflow and saturation occurs, the saturation logic loads the maximally positive 9.31 (0x7FFFFFFFFF) or maximally negative 9.31 value (0x8000000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user. This is referred to as 'super saturation' and provides protection against erroneous data or unexpected algorithm problems (e.g., gain calculations).

2. Bit 31 Overflow and Saturation:

3. When bit 31 overflow and saturation occurs, the saturation logic then loads the maximally positive 1.31 value (0x007FFFFFFF) or maximally negative 1.31 value (0x0080000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user. When this Saturation mode is in effect, the guard bits are not used (so the OA, OB or OAB bits are never set).

4. Bit 39 Catastrophic Overflow

The bit 39 overflow status bit from the adder is used to set the SA or SB bit, which remain set until cleared by the user. No saturation operation is performed and the accumulator is allowed to overflow (destroying its sign). If the COVTE bit in the INTCON1 register is set, a catastrophic overflow can initiate a trap exception.

- **Accumulator 'Write Back'**

The MAC class of instructions (with the exception of MPY, MPY.N, ED and EDAC) can optionally write a rounded version of the high word (bits 31 through 16) of the accumulator that is not targeted by the instruction into data space memory. The write is performed across the X bus into combined X and Y address space. The following addressing modes are supported:

1. W13, Register Direct:

The rounded contents of the non-target accumulator are written into W13 as a 1.15 fraction.

2. [W13] $\pm$ 2, Register Indirect with Post-Increment: The rounded contents of the non-target accumulator are written into the address pointed to by W13 as a 1.15 fraction. W13 is then incremented by 2 (for a word write).

- **Round Logic**

The round logic is a combinational block, which performs a conventional (biased) or convergent (unbiased) round function during an accumulator write (store). The Round mode is determined by the state of the RND bit in the CORCON register. It generates a 16-bit, 1.15 data value which is passed to the data space write saturation logic. If rounding is not indicated by the instruction, a truncated 1.15 data value is stored and the LS Word is simply discarded.

Conventional rounding takes bit 15 of the accumulator, zero-extends it and adds it to the ACCxH word (bits 16 through 31 of the accumulator). If the ACCxL

word (bits 0 through 15 of the accumulator) is between 0x8000 and 0xFFFF (0x8000 included), ACCxH is incremented. If ACCxL is between 0x0000 and 0x7FFF, ACCxH is left unchanged. A consequence of this algorithm is that over a succession of random rounding operations, the value will tend to be biased slightly positive.

Convergent (or unbiased) rounding operates in the same manner as conventional rounding, except when ACCxL equals 0x8000. If this is the case, the LS bit (bit 16 of the accumulator) of ACCxH is examined. If it is '1', ACCxH is incremented. If it is '0', ACCxH is not modified. Assuming that bit 16 is effectively random in nature, this scheme will remove any rounding bias that may accumulate.

The SAC and SAC.R instructions store either a truncated (SAC) or rounded (SAC.R) version of the contents of the target accumulator to data memory, via the X bus (subject to data saturation, see Section 2.4.2.4). Note that for the MAC class of instructions, the accumulator write back operation will function in the same manner, addressing combined MCU (X and Y) data space through the X bus. For this class of instructions, the data is always subject to rounding.

4.9.7 Data Space Write Saturation

In addition to adder/subtractor saturation, writes to data space may also be saturated, but without affecting the contents of the source accumulator. The data space write saturation logic block accepts a 16-bit, 1.15 fractional value from the round logic block as its input, together with overflow status from the original source (accumulator) and the 16-bit round adder. These are combined and used to select the appropriate 1.15 fractional value as output to write to data space memory.

If the SATDW bit in the CORCON register is set, data (after rounding or truncation) is tested for overflow and adjusted accordingly. For input data greater than 0x007FFF, data written to memory is forced to the maximum positive 1.15 value, 0x7FFF. For input data less than 0xFF8000, data written to memory is forced to the maximum negative 1.15 value, 0x8000. The MS bit of the source (bit 39) is used to determine the sign of the operand being tested.

If the SATDW bit in the CORCON register is not set, the input data is always passed through unmodified under all conditions.

4.9.8 BARREL SHIFTER

The barrel shifter is capable of performing up to 15-bit arithmetic or logic right shifts, or up to 16-bit left shifts in a single cycle. The source can be either of the two DSP accumulators or the X bus (to support multi-bit shifts of register or memory data).

The shifter requires a signed binary value to determine both the magnitude (number of bits) and direction of the shift operation. A positive value will shift the operand right. A negative value will shift the operand left. A value of 0 will not modify the operand.

The barrel shifter is 40 bits wide, thereby obtaining a 40-bit result for DSP shift operations and a 16-bit result for MCU shift operations. Data from the X bus is presented to the barrel shifter between bit positions 16 to 31 for right shifts, and bit positions 0 to 15 for left shifts.

4.10 CALCULATION

4.10.1 E-BICYCLE BATTERY

Cell: 3.6V 1.2A 11.5W

Battery Configuration

10 Series ($10 \times 3.6 = 36V$)

4 Parallel ($4 \times 2.5 = 10A$)

Battery capacity = 360W 36V 10 A

4.10.2 Panel:

Single Panel rating: 20W 18V 1.12A

Total number of Panels = 6

Panel Connection:

3 Parallel ($3 \times 1.12 = 3.36A$)

1 Series (18+18 = 36V)

Panel Output = 120W 36V 3.36A

SEPI Converter:

Output: 42V 2.5A

PV

BATTERY CURRENT=10AH

DC-DC CONVERTER FLOW CURRENT=2.5A/H

CHARGING TIME=BATTERY CURRENT/FLOW CURRENT

$$=10/2.5$$

$$=4 \text{ HOURS}$$

4.10.3 BRIDGELESS RECTIFIER:

Output: 42V 5A

GRID

BATTERY CURRENT=10A

AC-DC CONVERTER FLOW CURRENT=5A/H

CHARGING TIME=BATTERY CURRENT/FLOW CURRENT

$$=10/5$$

$$=2 \text{ HOURS}$$

4.11 EXPERIMENTAL OUTPUT:

4.11.1 CHECKING THE WEATHER CONDITION TO CHOOSE THE SOURCE OF CHARGING:

The system is designed in such a way that the sensor could sense the temperature of the day and if it is above 32°C, it will be displayed in LCD display “CONNECT TO SOLAR” and if the temperature of the day is below 32°C, it will be displayed in LCD display as “CONNECT TO GRID”. The accurate temperature can be monitored in blynk application.

4.11.2 PV CONNECTION



Fig 4.21 Blynk notification

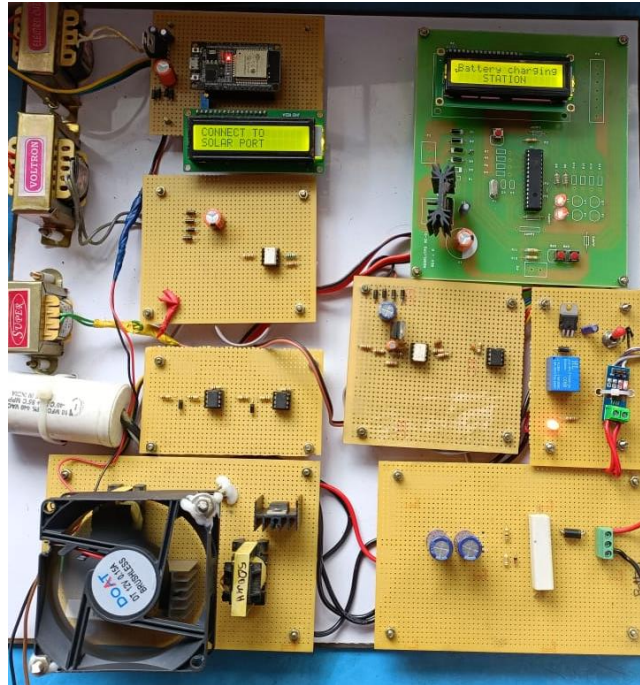


Fig 4.22 PV charging circuit

THE TEMPERATURE IS GREATER THAN 32 DEGREE SO WE ARE CHARGING THROUGH PV



Fig 4.23 Output of panel at maximum radiation

PV OUTPUT AT 2nd Floor

Voltage = 42.8V



Fig 4.24 PV output at ground floor

PV OUTPUT AT GROUND FLOOR

VOLTAGE AFTER LOSSES = 41.2V

4.11.4 CHARGING

HALF CHARGING TIME

FROM 10:30 am to 12:30 am



Fig 4.25 E-bicycle displayed in Half charged



Fig 4.26 E-bicycle charged through pv charging circuit

- We are connecting the solar panels(3 series with two parallel connection) 60W,36V to the SEPIC Converter.From the SEPIC Converter.This converter steps up the voltage to the sufficient amount and gives to the battery through safety circuit. Next to the converter we have connected the safety circuit and then connected to the battery.From the battery it starts to charge.
- We have designed the SEPIC Converter.MOSFET is used and the supply is given from the embedded
- For tuning the MOSFET embedded system is used. side.The supply for embedded system is given from Grid.
- From the grid the connection is given to transformer which steps down to 12V and then given to the rectifier to convert from AC to DC.From this 12V ,5V is taken by the microcontroller and then the switching circuit is controlled by MOSFET and the sepic converter is operated.

4.11.5 GRID CONNECTION

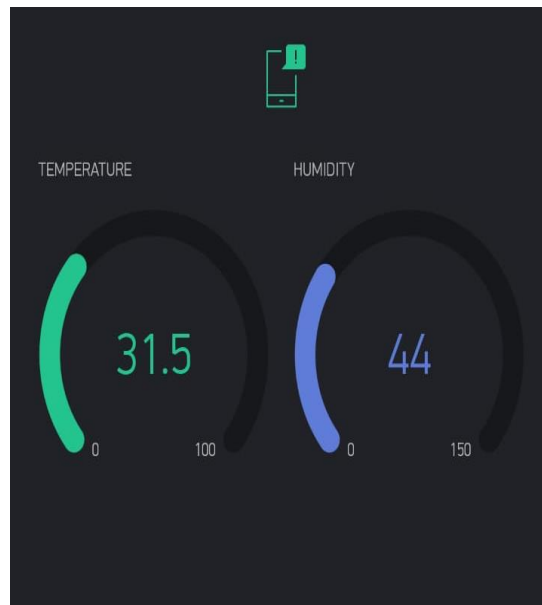


Fig 4.27 Blynk notification

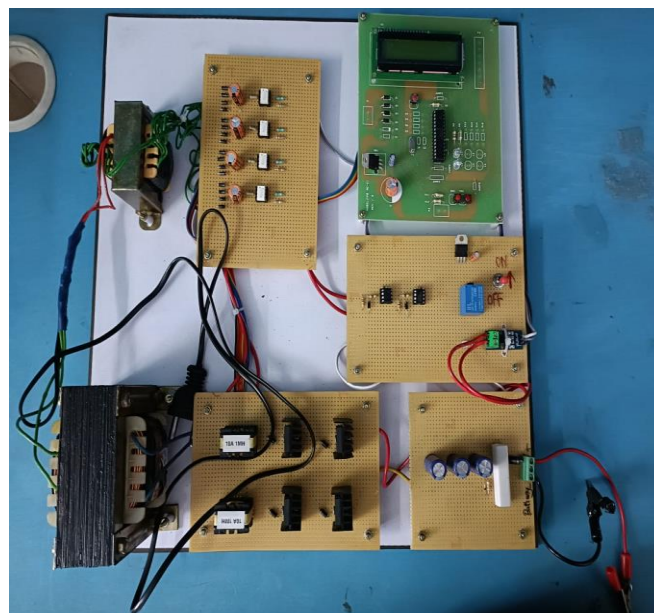


Fig 4.28 Grid charging circuit

THE TEMPERATURE IS LESSER THAN 32 DEGREE SO WE ARE CHARGING THROUGH GRID.

The supply is taken directly from the grid. Two transformers are used. One steps down from 230V to 12V and the another steps down from 230V to 24V. We have used microcontroller, bridge rectifier(AC to DC). Then this DC power is given as

supply to the microcontroller. Bridgeless rectifier is used here made with MOSFET. Input is given from the second transformer which steps down as 24V. Output from bridgeless rectifier is given to the safety circuit and then to the battery. It starts to charge the vehicle. For the microcontroller we have used Embedded C to measure the temperature, humidity, current sensor and display.

IOT is implemented to check the amount of all the above factors and displays whether it would be able to charge the vehicle in solar pv or through grid. MATLAB (Simulink) is used for tuning the MOSFET in SEPIC converter and bridgeless rectifier.



Fig 4.29 E-bicycle charged by Grid

4.11.6 CHARGING

HALF CHARGING TIME:

FROM 4PM – 5 PM



Fig 4.30 half charged by grid displayed by E-bicycle

4.12 FEATURES:

- To reduce the burden on the grid and use of EVs at remote locations, a renewable energy source based off-grid EV charging station is introduced.
- The reliability of the off-grid EV charging station is improved by using energy storage system.
- Different sources available at hybrid grid, and able to deliver stable energy needed.
- High Efficiency
- Power balanced
- Power management
- Power consumption
- To reduce the burden on the grid and use of EVs at remote locations, a renewable energy source based off-grid EV charging station is introduced.
- The reliability of the off-grid EV charging station is improved by using energy storage system.

- The charging and discharging of the energy storage system is governed on the basis of PV irradiance

CHAPTER 5

CONCLUSION

5.1 CONCLUSION

In this project, a charging station (HGCS) is proposed to charge the EV battery. This minimizes the grid burden and increases the EVs utilization at remote locations by using the PV. In this project, an ESS is connected to the OGCS which make the system work in any circumstances. The OGCS with ESS exchange the power to charge the EV battery during the absence or reduced sunlight. Further, a constant current method is used to charge the EV battery at various C-rate. Altogether, this paper presents a more sustainable and efficient OGCS as well as favours pollution-free transportation.

REFERENCE

- [1] Li X, Hui D and Lai X. Battery energy storage station (bess)-based smoothing control of photovoltaic (pv) and wind power generation fluctuations. IEEE Transactions on Sustainable Energy, 4(2):464–473, April 2013. ISSN 1949-3029.
- [2] Nordin A. H. M. and Omar A. M.. Modeling and simulation of photovoltaic (pv) array and maximum power point tracker (mppt) for grid-connected pv system. In 2011 3rd International Symposium Exhibition in Sustainable Energy Environment (ISESEE), pages 114–119, June 2011.
- [3] Pinak J. Tulpule, Vincenzo Marano, Stephen Yurkovich, and Giorgio Rizzoni. Economic and environmental impacts of a pv powered workplace parking garage charging station. Applied Energy, 108:323 – 332, 2013. ISSN 0306-2619.
- [4] Singh M, Kumar P, Kar I, and Kumar N. A real time smart charging station for evs designed for v2g scenario and its coordination with renewable energy sources. In 2016 IEEE Power and Energy Society General Meeting (PESGM), pages 1–5, July 2016.

- [5] <https://ieeexplore.ieee.org/document/7947625> [Smart grid integration of micro hybrid power system using 6-switched 3-level inverter]
- [6] <https://ieeexplore.ieee.org/document/5484254> [Plug-in Hybrid Electric vehicles in the smart grid environment]
- [7] <https://ieeexplore.ieee.org/document/8567002> [Multi objective optimization of Grid- Photovoltaic- Electric Vehicle Hybrid system in Smart Building with Vehicle-to-Grid (V2G) concept]
- [8] <https://ieeexplore.ieee.org/document/9362768> [Multi-loop Multi-objective Control of Solar Hybrid EV Charging Infrastructure for Workplace]