POWER OPTIMIZER FOR MPPT SOLAR PV SYSTEM

A PROJECT REPORT

Submitted by

ABHINAV RAJAGOPALAN	**212713105001**
ALAGESWARAN V	**212713105007**
NAREN SRIVATSAV G	**212713105058**
DHAMOTHARAN S	**212713105301**

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

SRI VENKATESWARA COLLEGE OF ENGINEERING, PENNALUR, SRIPERUMBUDUR TALUK

ANNA UNIVERSITY : CHENNAI 600 025

APRIL 2017

ANNA UNIVERSITY : CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report **“ POWER OPTIMIZER FOR SOLAR MPPT PV SYSTEM ” is the bonafide work of “ABHINAV RAJAGOPALAN, ALAGESWARAN V, NAREN SRIVATSAV G, DHAMOTHARAN S”** who carried out the project work under my supervision.

SIGNATURE
Dr.K.R. Santha, Ph.D.
HEAD OF THE DEPARTMENT
Professor and Head,
Electrical and Electronics Engineering
Sri Venkateswara College of Engineering
Post Bag No.1,
Pennalur,
Sriperumbudur Taluk – 602 117

SIGNATURE
Dr. A.Tamizhselvan, M.E,
SUPERVISOR
Assistant Professor
Electrical and Electronics Engineering
Sri Venkateswara College of Engineering
Post Bag No.1,
Pennalur,
Sriperumbudur Taluk – 602 117

Submitted for the Viva Voce Examination held on ____________________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We are highly indebted to **Dr. Ganesh Vaidyanathan**, Principal, Sri Venkateswara College of Engineering, for providing us an opportunity to work on this project.

We would like to express our sincere thanks to **Dr.K.R. Santha**, Head of the Department, Department of Electrical and Electronics Engineering, Sri Venkateswara College of Engineering, for her valuable support and suggestions for carrying out our project work.

We would like to express our special gratitude and thanks to our respectful guide **Mr.A. Tamizhselvan**, Assistant Professor, Department of Electrical and Electronics Engineering, Sri Venkateswara College of Engineering, for his kind encouragement and support throughout this project.

We would like to express our sincere thanks to our Project Coordinators **Dr.T. Annamalai**, Associate Professor Department of Electrical and Electronics Engineering, Sri Venkateswara College of Engineering, for their constant support given during this project.

We also would like to convey our sincere thanks to all the other faculty members of the department for their time and attention.

ABSTRACT

A DC-DC converter is an electronic circuit that converts a source of DC from one voltage level to another. Converters developed to maximize the energy harvest for PV systems are called power optimizers. Maximum power point tracker (MPPT) is basically an electronic system that controls the duty circuit of the converter to enable the photovoltaic module operate at maximum operating power at all condition and not some sort of mechanical tracking system that physically rotate the photovoltaic modules to face sunlight directly. To extract maximum power, you must adjust the load to match the current and voltage of the solar panel. The converter must be designed to be connected directly to the photovoltaic panel and perform operation to search the maximum power point (MPPT). Power can be maintained constant with the help of proposed Luo converter by properly controlling the duty cycle of the MOSFETs. It requires only two switches and by properly controlling the switches, buck or boost mode can be achieved. The MATLAB simulated model of the solar panel followed by the dc–dc converter is presented and waveforms obtained are discussed. The dc to dc converter model is programmed in MPPT mode using optimal duty ratio to achieve maximum output. The performance of the complete system model under varying insolation levels of solar panel is discussed.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

D	DUTY RATIO
ΔD	DUTY RATIO STEP
H	SWITCHING FREQUENCY
$\mathbf{I_S}$	SOURCE CURRENT
K	KELVIN
P	POWER
R	RESISTANCE
$\mathbf{R_{int}}$	INTERNAL RESISTANCE
$\mathbf{R_L}$	LOAD RESISTANCE
T	TEMPERATURE RCI
$\mathbf{T_C}$	COLD SIDE TEMPERATURE
$\mathbf{T_H}$	HOT SIDE TEMPERATURE
$\mathbf{T_S}$	SWITCH PERIOD
$\mathbf{T_{OFF}}$	OFF TIME OF SWITCH
$\mathbf{T_{ON}}$	ON TIME OF SWITCH
$\mathbf{T_S}$	ALGORITHM UPDATE PERIOD
ΔT	TEMPERATURE DIFFERENCE
U	VOLTAGE
$\mathbf{U_{BAT}}$	BATTERY VOLTAGE
$\mathbf{U_{IN}}$	INPUT VOLTAGE
$\mathbf{U_O}$	OUTPUT VOLTAGE
$\mathbf{U_{OL}}$	OPENLOAD VOLTAGE
$\mathbf{U_{PWM}}$	PULSE WIDTH VOLTAGE
$\mathbf{U_{REF}}$	REFERENCE VOLTAGE
$\mathbf{U_{TRI}}$	TRIANGLE VOLTAGE
A	SEEBECK COEFFICIENT
ZT	FIGURE OF MERIT

CHAPTER 1

SOLAR CELLS

1.1 INTRODUCTION

Solar cells and photodetectors are devices that convert an optical input into current. A solar cell is an example of a photovoltaic device, i.e, a device that generates voltage when exposed to light. The photovoltaic e ect was discovered by Alexander-Edmond Becquerel in 1839, in a junction formed between an electrode (platinum) and an electrolyte (silver chloride). The rst photovoltaic device was built, using a Si pn junction, by Russell Ohl in 1939. The functioning of a solar cell is similar to the photodiode (photodetector). It is a photodiode that is unbiased and connected to a load (impedence). There are three qualitative differences between a solar cell and photodetector.

1. A photodiode works on a narrow range of wavelength while solar cells need to work over a broad spectral range (solar spectrum).
2. Solar cells are typically wide area devices to maximize exposure.
3. In photodiodes the metric is quantum e efficiency, which de nes the sig-nal to noise ratio, while for solar cells, it is the power conversion e - efficiency, which is the power delivered per incident solar energy. Usually, solar cells and the external load they are connected to are designed to maximize the delivered power.

1.2 SOLAR SPECTRUM

The solar spectrum typically extends from the IR to the UV region, wave-length range from 3 m to 0.2 m. But the intensity is not uniform. A typical solar spectrum, as a plot of spectral irradiance vs. wavelength, is shown in figure 1. The area under the curve gives the total areal intensity and this is approximately 1.35 kW m ^{2}. The solar spectrum can be approximated by a black body radiation curve at temperature of approximately 5250 C. There is also a difference in the spectra measured at the top of the atmosphere and at the surface, due to atmospheric scattering and absorption.

The path length of the light in the atmosphere depends on the angle, which will vary with the time of day. This is given by the air mass number (AM), which is the secant of the angle between the sun and the zenith (sec). AM0 represents the solar spectrum outside the earth's atmosphere. AM1 is when the angle is zero, i.e. sun is at the zenith and it has an intensity of 0.925 kW m ^{2}. AM2 is when sun is at angle of 60 and its intensity is 0.691 kW m ^{2}. The spectra are plotted in figure.

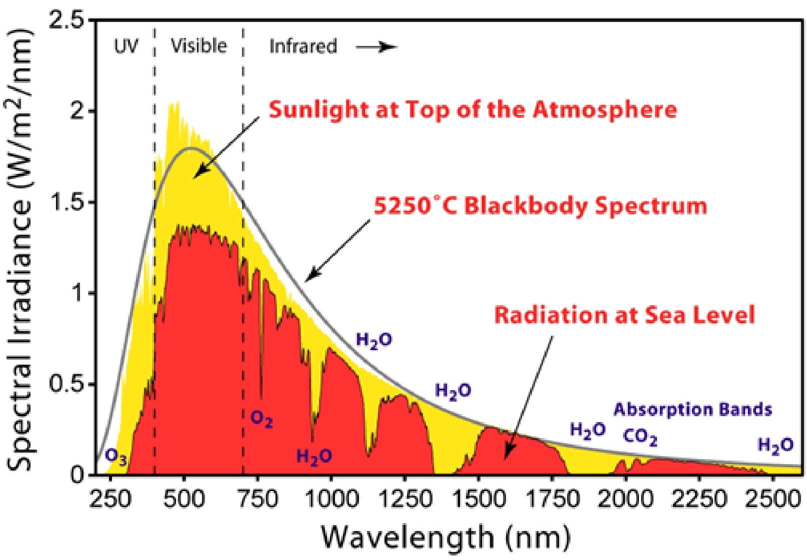


Figure 1: Typical solar spectrum

1.3 SOLAR CELL WORKING PRINCIPLE

A simple solar cell is a pn junction diode. The schematic of the device is shown in figure 4. The n region is heavily doped and thin so that the light can penetrate through it easily. The p region is lightly doped so that most of the depletion region lies in the p side. The penetration depends on the wave-length and the absorption coe cient increases as the wavelength decreases. Electron hole pairs (EHPs) are mainly created in the depletion region and due to the built-in potential and electric eld, electrons move to the n region and the holes to the p region. When an external load is applied, the excess elec-trons travel through the load to recombine with the excess holes. Electrons and holes are also generated with the p and n regions, as seen from figure 4. The shorter wavelengths (higher absorption coe cient) are absorbed in the n region and the longer wavelengths are absorbed in the bulk of the p re-gion. Some of the EHPs generated in these regions can also contribute to the current. Typically, these are EHPs that are generated within the minoritycarrier di usion length, L_e for electrons in the p side and L_h for holes in the n side. Carriers produced in this region can also di use into the depletion region and contribute to the current. Thus, the total width of the region that contributes to the solar cell current is $w_d + L_e + L_h$, where w_d is the depletion width. This is shown in figure 5. The carriers are extracted by metal electrodes on either side. A nger electrode is used on the top to make the electrical contact, so that there is su cient surface for the light to penetrate. The arrangement of the top electrode is shown in figure 6.

Consider a solar cell made of Si. The band gap, E_g, is 1.1 eV so that wave-length above 1.1 m is not absorbed since the energy is lower than the band gap. Thus any greater than 1.1 m has negligible absorption. For much smaller than 1.1 m the absorption coe cient is very high and the EHPs are generated near the surface and can get trapped near the surface defects.

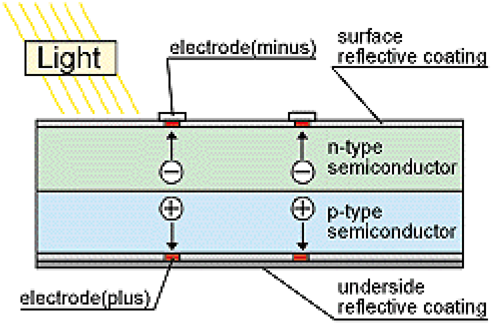


Figure 2: Principle of operation of a pn junction solar cell.

1.4 SOLAR CELL I-V CHARACTERISTICS

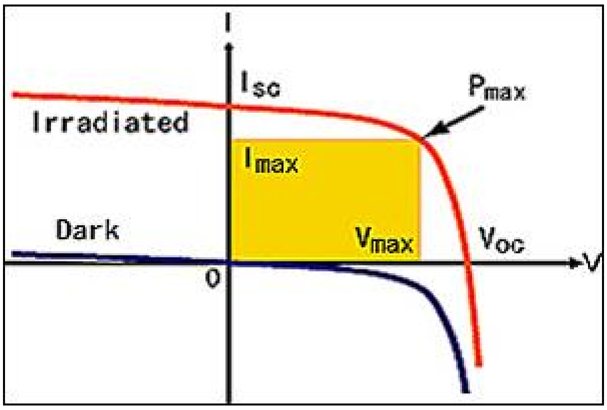


Figure 3: I-V curve for a solar cell with maximum power indicated by the shaded area.

It possible to calculate the I-V characteristics of the solar cell by considering its equivalent circuit. The I-V characteristics depend on the intensity of the incident radiation and also the operating point (external load) of the cell. Consider a pn junction solar cell under illumination, as shown in figure 7. If the external circuit is a short circuit (external load resistance is zero) then the only current is due to the generated EHPs by the incident light. This is called the photocurrent, denoted by I_{ph}. Another name for this is the short circuit current, I_{sc}. By de nition of current, this is opposite to the photo current and is related to the intensity of the incident radiation, I_{op}, by

$$I_{sc} = I_{ph} = kI_{op} \qquad (1)$$

where k is a constant and depends on the particular device. k is equivalent to an e efficiency metric that measures the conversion of light into EHPs. Consider the case when there is an external load R, as shown in figure 7. The equivalent circuit for this case is shown in figure 8. There is a voltage across the external load, given by V = IR. This voltage opposes the built in potential and reduces the barrier for carrier injection across the junction. This is similar to a pn junction in forward bias, where the external bias causes injection of minority carriers and increased current. This forward bias current opposes the photo current generated within the device due to the solar radiation.

$$I = I_{ph} + I_d$$

$$I_d = I_{s0} \left[\exp\left(\frac{eV}{k_B T}\right) \quad 1\right] \qquad (2)$$

$$I = I_{ph} + I_{s0}\left[\exp\left(\frac{eV}{k_B T}\right) \quad 1\right]$$

This is because I_{ph} is generated due to electrons going to the n side and holes to the p side due to the electric eld within the device, i.e. drift current while the forward bias current is due to diffusion current caused by the injection of minority carriers.

Where I_d is the forward bias current and can be written in terms of the reverse saturation current, I_{s0} and external voltage, V. The overall I-V characteristics is plotted in figure.

$$I_{ph} = I_{so}\ [\exp(eV_{oc}/k_BT)] \qquad (3)$$

$$V_{oc} = K_BT/e\ \ln\ [I_{PH}/I_{so}]$$

In the absence of light, the dark characteristics is similar to a pn junction I-V curve. The presence of light (I_{ph}) has the e ect of shifting the I-V curve down. From figure 9, it is possible to de ne a photo current I_{ph}, which is the current when the external voltage is zero and an open circuit voltage, V_{oc}, which is the voltage when the net current in the circuit is zero. Using equation 2, V_{oc} can be calculated as.

Higher the photon ux, higher is the value of I_{ph} (by equation 1), and higher the value of V_{oc}. Similarly, lower I_{s0} can also cause higher V_{oc}. Since I_{s0} is the reverse saturation current for the pn junction it is given by

$$I_{s0} = n_i^2 e\ \left[\frac{D_e}{L_eN_A} + \frac{D_h}{L_hN_D}\right] \qquad (4)$$

The reverse saturation current can be lowered by choosing a material with a higher band gap, E_g, which will cause n_i to be lower. But this will also reduce the range of wavelengths that can be absorbed by the material, which will have the e ect of lowering I_{ph}.

The total power in a solar cell is given by

$$P = IV = I_{so}\, V[\ \exp\ (K_BT)]^{Ev}\ I_{ph} \qquad (5)$$

For maximum power, its derivative with respect to voltage should be zero. This gives a recursive relation in current and voltage.

$$\frac{dP}{dV} = 0$$

$$Im = Iph \left(1 + \frac{Kb}{eV} T \right)$$

$$Vm = VocKbT \ln\left(1 + e \quad \frac{eVm}{KbT} \right)$$

$$Pm = ImVm = Iph[Voc \quad e \ln \left(1 + \frac{eVm}{KbT} \right) \qquad (6)$$

This can be seen from figure 10. The area under the curve, corresponding to I_m and V_m, gives the maximum power. From equation 6 it can be seen that the maximum power is directly proportional to V_{oc} and can be increased by decreasing I_{s0}. This means that smaller n_i and a larger E_g are favorable but the trade o is that less radiation is absorbed.

1.5 SOLAR CELL MATERIALS AND EFFICIENCY

Conventional solar cells are made of Si single crystal and have an e efficiency of around 22-24%, while polycrystalline Si cells have an e efficiency of 18%. A schematic representation of such a cell is shown in figure 6. The e efficiency of the solar cell depends on the band gap of the material and this is shown in figure 11. Polycrystalline solar cells are cheaper to manufacture but have a lower e efficiency since the microstructure introduces defects in the material that can trap carriers. Amorphous solar cells have an even lower e efficiency but can be grown directly on glass substrates by techniques like sputtering so that the overall cost of manufacturing is lowered. There are also design improvements in the solar cell that can enhance the e efficiency. PERL (passivated emitter rear locally di used) cells, shown in figure 12, have an e efficiency of 24% due to the inverted pyramid structure etched on the surface that enhances absorption.

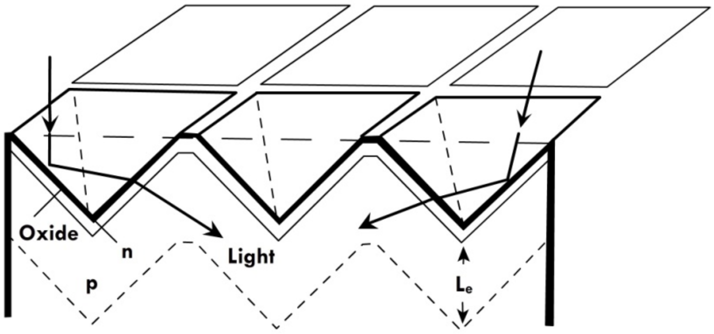


Figure 4: Si solar cell with an inverted pyramid structure

Typical solar cells are made of the same material so that the pn junction is a homojunction. Some solar cell materials and their e ciencies are summa-rized in table 1. A comprehensive state of current research in di erent solar cell technologies and their e efficiency is available in figure 13. Heterojunction solar cells are also possible and they have the advantage of minimizing ab-sorption in regions other than the depletion region, but overall cost increases because of the use of di erent materials and the tight processing conditions needed to produce defect free interfaces. A schematic of such a cell based on GaAs/AlGaAs is shown in figure 14. The shorter wavelengths are absorbed by the AlGaAs layers while the longer wavelengths, with higher penetration depths, are absorbed by the GaAs layer. This leads to an overall e efficiency of around 25%, see table 1. It is also possible to have a homojunction solar cell but with a passivating layer of another material at the surface to reduce defects. This is shown in figure 15. The surface passivating layer removes the dangling bonds and minimizes carrier trapping. The passivation layer is a thin layer of a higher band gap material to minimize absorption. Similarly, amorphous semiconductor materials like Si and Ge also have a passivating layer of H, a-Si:H or a-Ge:H, to reduce dangling bonds.

Another way of improving solar cell efficiency is to have more than one cell in tandem. These are called tandem solar cells and a schematic is shown in figure 16. These consist of two pn junction solar cells, with the rst one having a higher band gap than the second. Thus, the shorter wavelengths can be absorbed in cell 1, see figure 16, while the longer wavelengths are absorbed in cell 2. The advantage is that a larger portion of the solar radiation is used so that tandem cells have high e efficiency, see table1, but it also adds a layer of complexity in growth and increases cost. Tandem cells can also be made using amorphous Si:H and Ge:H. These are cheaper to make and more efficient than individual amorphous solar cell devices.

Table 1: Some common solar cell materials and their characteristics.

Semi-conductor	E_g (eV)	V_{oc} (V)	J_{sc} (mA cm ^{2})	(%)	Commen-Ts
Si, single crystal	1.1	0.5-0.7	42	16-24	Single crystal, PERL
Si, poly-crys-Talline	1.1	0.5-0.65	38	12-19	
Amorphous Si:Ge:H lm				8-13	Amorphous lms with Tandem structure, large-area Fabrication
GaAs, Single crystal	1.42	1.02	28	24-25	

GaAlAs/ GaAs, tandem		1.03	27.9	25	Di erent band gap Materials in tandem Increases absorp- Tion e efficiency
GaInP/ GaAs, tandem		2.5	14	25-30	Di erent band gap Materials in tandem Increases absorp- Tion e efficiency
CdTe, thin lm	1.5	0.84	26	15-16	
InP, single Crystal	1.34	0.87	29	21-22	
$CuInSe_2$	1.0			12-13	

CHAPTER 2

CONVERTERS

2.1 CONVERTER INTRODUCTION:

The task of a power converter is to process and control the flow of electrical energy by supplying voltages and currents in a form that is optimally suited for user loads.

Energy conversions were initially achieved using electromechanical converters (which were mainly rotating machines). Today, with the development and the massive production of power semiconductors, static power converters are used in numerous application domains and especially in particle accelerators. Their weight and volume are smaller and their static and dynamic performance are better.

A static converter is composed of a set of electrical components building a meshed network that acts as a linking, adapting, or transforming stage between two sources, generally between a generator and a load (Fig. 5).

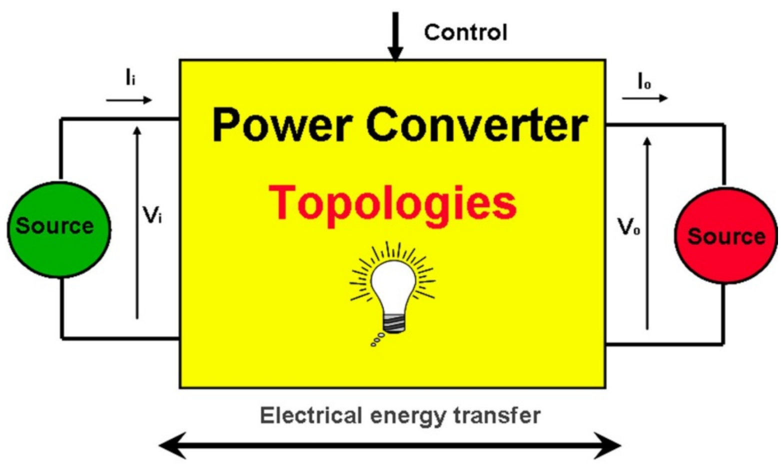


Figure 5:Converters Topology

2.2 ELEMENTS OF A CONVERTER:

An ideal static converter allows control of the power flow between the two sources with 100% efficiency. A large part of power converter design is the optimization of its efficiency. But as a first approach and to define basic topologies, it is interesting to take the hypothesis that no losses occur through a power converter's conversion process. With this hypothesis, the basic elements are of two types:

- non-linear elements that are, most of the time, electronic switches: semiconductors used in commutation mode [1];
- linear reactive elements: capacitors, inductances (and mutual inductances or transformers). These reactive components are used for intermediate energy storage but also perform voltage and current filtering. They generally represent an important part of the size, weight and cost of the equipment [2, 3].

2.3 LUO CONVERTER TYPES:

Voltage Lift (VL) Technique is a popular method widely used in electronic circuit design. It has been successfully employed in DC/DC converter applications in recent years and opened a way to design high voltage gain converters. Four series Luo-Converters [1-9] are the examples of VL technique implementations. However, the output voltage increases in stage by stage just along the arithmetic progression [10]. Luo series Re-Lift and Super lift converters introduces a novel approach – Super-Lift (SL) technique that implements the output voltage increasing in stage by stage along the geometric progression. It effectively enhances the voltage transfer gain in power series.

In order to sort these converters different from existing VL converters, we entitle converters "Positive Output Super-lift converters". These are two sub-series: main series and additional series. Each circuit of the main series has one switch S, n inductors, 2n capacitors and (3n -1) diodes. Each circuit of the additional series has one switch S, n inductors. 2(n+1) capacitors and (3n+1) diodes. The conduction duty ratio is k, switching frequency is f, switching period is T=1/f, the load is resistive load R. The input voltage and current are V_{in} and I_{in}, output voltage and current are Vo and Io. Assume no power losses during the conversion process, V_{in} x I_{in}=V_o x

I_o. The voltage transfer gain is G: G = $\frac{V0}{}$

Here we introduce the first three stages of Positive Output Super-Lift Converters. For convenience to explain, we call them Elementary circuit, Re-Lift circuit and Triple-Lift circuit respectively. We can number them as n=1, 2 and 3.

2.3.1 SUPER LIFT LUO CONVERTER ELEMENTORY CIRCUIT

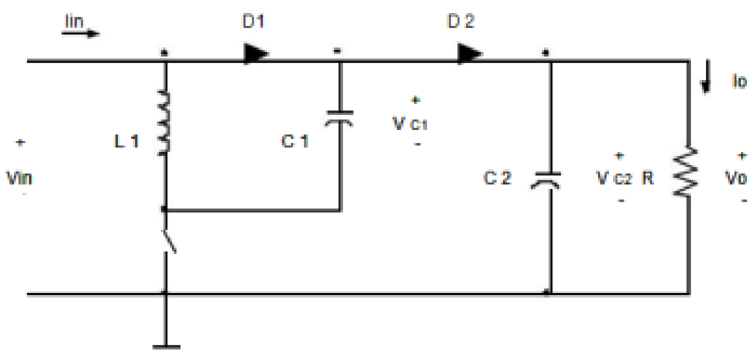


Figure 6: Elementary circuit

The Voltage across capacitor C_1 is charged to V_{in}. The current i_{L1} flowing through inductor L_1 increases with voltage V_{in} during switching-on period kT and decreases with voltage – (V_o –2V_{in}) during switching-off period (1-k)T.

2.3.2 RE-LIFT CIRCUIT

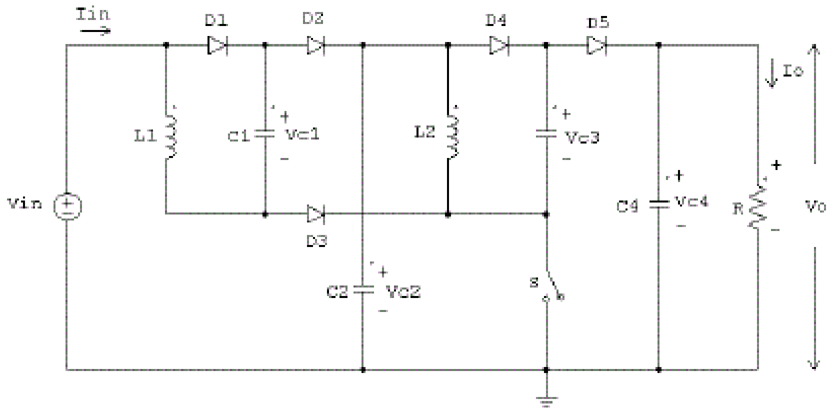


Figure 7:Relift circuit

The Re-Lift circuit is derived from Elementary circuit by adding the parts (L_2-D_3-D_4-D_5-C_3-C_4). Its circuit diagram is shown in figure.

The voltage across capacitor C_1 is charged to V_{in}. As described in previous section the voltage V_1 across capacitor.

The voltage across capacitor C_3 is charged to V_1. The current flowing through inductor L_2 increases with Voltage V_1 during switching-on period k and decreases with voltage $-(V_o-2V_1)$ during switching-off period (1-k)T.

2.3.3 TRIPLE-LIFT CIRCUIT

Triple-Lift circuit is derived from Re-Lift circuit by secondly repeating the parts (L_2-D_3-D_4-D_5-C_3-C_4). Its circuit diagram is shown in figure 3 below. The voltage across capacitor C_5 is charged to V_2. The current flowing through inductor L_3 increases with voltage V_2 during switching-on period and decreases with voltage $-(V_o-2V_2)$ during switching-off (1-k) T.

2.3.4. HIGHER ORDER LIFT CIRCUIT

Higher order Lift circuit can be designed by just multiple repeating the parts (L_2-D_3-D_4-D_5-C_3-C_4).

2.4 MODES OF OPERATION:

2.4.1. MODE 1:

In mode one of operation in a superlift luo converter

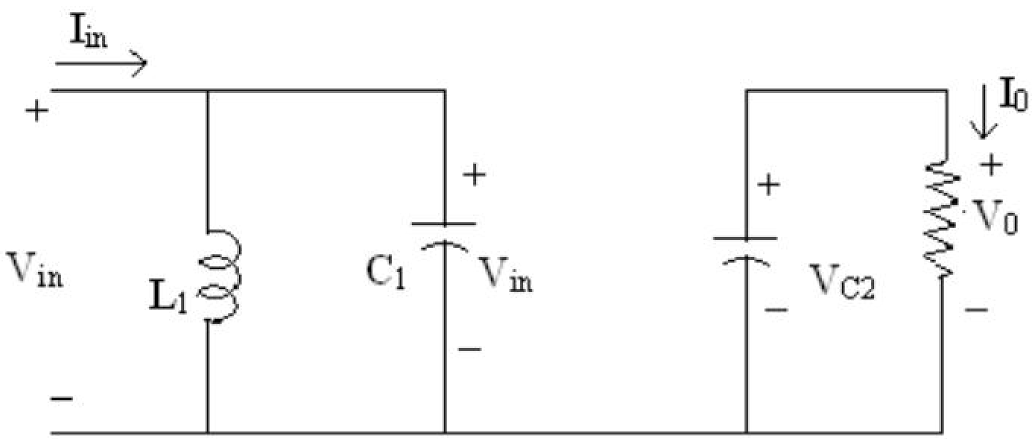


Figure 8: Mode I

The switch is in the ON state and the dc output transmitted directly to the inductor L and capacitor C1.In this mode the voltage across capacitor C1 is charged to Vin when switch S is in on position. Because inductor L and capacitor C1 are connected in parallel, the current IL1 will increase with voltage Vin.The inductor charges to the voltage of

$$VinkTie\ \Delta I=Vinkt/L \quad (1)$$

2.4.2 MODE 2:

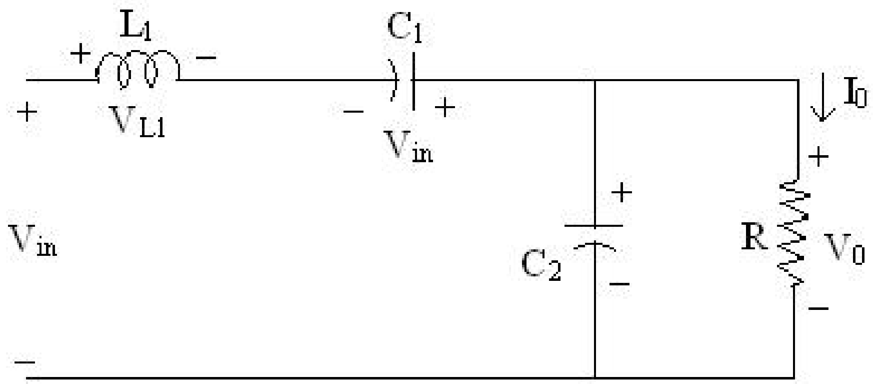


Figure 9: Mode II

In mode 2, the switch is turned OFF, during this state, the voltage across inductor L will become

$$(Va -2Vin)\ \Delta I= -(Va -2Vin)\ (1-kT)/L \qquad (2)$$

So the current IL1 will decrease. It is assumed that kT is the switch-on period and (l-k)T is the switch-off period.

Equating equations (1)&(2)

We can obtain the following equation :

$$VinkT + [-(Vo - 2Vin)(1-k)T] = 0 \qquad (3)$$

Then the output voltage can be calculated from the above formula: =

$$Vo = 2-k/1-kVin \qquad (3a)$$

Voltage transfer gain is:

$$G = Vo/Vin = 2-k/1-k \qquad (3b)$$

Iin-off = IL1-off = Ic1-off

Iin-on = IL1-on + Ic1-on

Considering T = 1/f and Vin/Iin = [(1-k)/(2-k)]^2 R

The variation ratio of inductor current IL1 is

$$i_{in-off} = i_{C1-off} = I_{L1}$$

$$i_{in-on} = i_{L1} + \frac{1-k}{k} I_{L1} = \frac{1_{L1}}{k}$$

$$i_{C1-on} = \frac{1-K}{K} = I_{L1}$$

And average input current

$$i_{in} = K i_{in-on} + (1-k) i_{in-off} = I_{L1} + (1-k) I_{L1} = (2-k) I_{L1} \quad (4)$$

Considering

$$\frac{V_{in}}{I_{in}} = \left(\frac{1-k}{2-k}\right)^2 \frac{V_o}{I_o} = \left(\frac{1-k}{2-k}\right)^2 R \quad (5)$$

The variation ratio of current i_{L1} through inductor L_1 is

$$\xi_1 = \frac{\Delta i_{L1}/2}{I_{L1}} = \frac{k(2-k)TV_{in}}{2L_1 I_{in}} = \frac{k(1-k)^2}{2(2-k)} \frac{R}{fL_1} \quad (6)$$

Usually €1 is small (much lower than unity); it means this converter normally works in the continuous mode.

The ripple voltage of output voltage V_O is

$$\Delta v_o = \frac{\Delta Q}{C_2} = \frac{I_o kT}{C_2} = \frac{k}{fC_2} \frac{V_o}{R} \quad (7)$$

Therefore, the variation ratio of output voltage Vo is

$$\varepsilon = \frac{\Delta v_o / 2}{V_o} = \frac{k}{2 RfC_2} \quad (8)$$

Usually R is in kΩ, f in 10 kHz and C_2 in μF, this ripple is very smaller than 1%

2.5 MODELLING OF SUPER LIFT LUO CONVERTER:

2.5.1 Modeling Methods

Modeling is the representation of physical behavior by mathematical means. The Simplified models yield physical insight, allow in design system to operate in specified manner. After basic insight has been gained, model can be refined to account for some of the previously neglected phenomenon. The most commonly used modeling methods are: State space averaging: Matrix based approach which gives insight into quantitative nature of basic averaging approximation and Circuit averaging technique:

Method based on equivalent circuit manipulations, resulting in a single equivalent linear circuit model of power stage.

2.5.2 STATE SPACE MODELLING:

Modelling of Elementary Luo Converter Using State Space:

Apply KVL Determine A, B, C, E Matrices

Figure 1 shows elementary Luo converter. In this positive output Luo converter, there are two states i.e., when switch is on and when switch is off. During each state we write the following

Equation 1 and 2:

x ' = A1x +B1Vd during dTs (1)

x'= A2x + B2Vd during(1-d)Ts (2)

where, A_1+ A_2 are state matrices and B_1 and B_2 are vectors. Let X_1 be inductor current, X_2 represent voltage across capacitor 1, X_3 represent voltage across capacitor 2. Figure 2 and 3 shows the turn on and turn off

For turn on:

$$\begin{pmatrix} x1' \\ x2' \\ x3' \end{pmatrix} = \begin{pmatrix} 0 & 1/L & 0 \\ -1/C1 & -1/(RiC1) & 0 \\ 0 & 0 & -1/RC2 \end{pmatrix} \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix} + \begin{pmatrix} 0 \\ 1/RiC1 \\ 0 \end{pmatrix} Vd \qquad (3)$$

FOR TURN OFF:

$$\begin{pmatrix} x1' \\ x2' \\ x3' \end{pmatrix} = \begin{pmatrix} 0 & 1/L & 1/L \\ -1/C1 & 0 & 0 \\ -1/C2 & 0 & -1/RC2 \end{pmatrix} \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix} + \begin{pmatrix} 1/L \\ 0 \\ 0 \end{pmatrix} Vd \qquad (4)$$

$$A = A1D + A2\,(1-D) \qquad (5)$$

$$B = B1D + B2(1-D) \qquad (6)$$

$$A = \begin{pmatrix} 0 & 1/L & -(1-D)/L \\ -1/C1 & -D/(RIC1) & 0 \\ (1-D)/C2 & 0 & -1/RC2 \end{pmatrix} \quad B = \begin{pmatrix} (1-D)/L \\ D/RIC1 \\ 0 \end{pmatrix} \quad C=(0\ 0\ 1) \qquad (7)$$

USING LAPLACE TRANSFORMATION:

$$SX^{\wedge}(S) = AX^{\wedge}(S) + [(A1-A2)X + (B1 - B2)V]D(S) \qquad (8)$$

$$X^{\wedge}(S) = [SI - A]^{-1}[(A1-A2)X + (B1 - B2)V]D(S) \qquad (9)$$

$$\underline{V_0^{\wedge}(S)} = C[SI - A]^{-1}[(A1 - A2)X + (B1 - B2)VD] + (C1 - C2)X\,D(S) \qquad (10)$$

$$SI - A = \begin{pmatrix} S & -1/L & (1-D)/L \\ -1/C & S+D/(R_1C_1) & 0 \\ -(1-D)/C_2 & 0 & S+1/(RC_2) \end{pmatrix} \qquad (11)$$

REMOVING R_I TERMS

$\underline{V_0}^{\wedge} = \quad V \quad \underline{-SL(2-D)} \quad +R$

$$D \quad \frac{(1-D)^2}{LRC_2 + SL + R(1-D)} \qquad (12)$$

2.5.3 STATE SPACE AVERAGING TECHNIQUE:

IN ON CONDITION

$$V_L(T) = V_G(T) \qquad (13)$$

$$IC_1(T) = I_G(T) - I(T) \qquad (14)$$

$$IC_2(T) = \frac{-V_0}{R} \qquad (15)$$

OFF CONDITION:

$$V_L(T) = 2V_G(T) - V_0(T) \qquad (16)$$

$$IC_1(T) = I_G(T) \qquad (17)$$

$$IC_2(T) = I_G(T)\frac{V_0}{R} \qquad (18)$$

$$V_L(T) = DV_G(T) + D^1(2V_G(T) - V_0(T) \qquad (19)$$

$$IC_1(T) = D(I_G(T) \pm I(T)) + D^1(I_G(T)) \qquad (20)$$

$$C_2(T) = D \quad -V_0/R \quad + \quad 1/D(I_G(T)) \quad -V_0/R \qquad (21)$$

INTRODUCING PERTUBATIONS:

$$V'(T) = V_G + V_G(T) \qquad (22)$$

$$V_0(T)=V_0 + V_0^{\wedge}(T) \quad (23)$$

$$I_G(T) = I_G + I_G(T) \quad (24)$$

$$I(T) = I + I(T) \quad (25)$$

ADDING PERTUBATIONS AND ELIMINATING DC TERMS:

$$SI^{\wedge}(S) = (2-D)V_G^{\wedge}(S) + D^{\wedge}(S) + D^{\wedge}(S)V_0 - V_G \; -DV^{\wedge}(S) \quad (26)$$

$$C_1SV^{\wedge}_G(S) = I_G(S) - DI(S) - D(S)I \quad (27)$$

$$C_2SV^{\wedge}_0(S) = DI_G(S) - DI(S) - D(S)I_G - \frac{V^{\wedge}_0(S)}{R} \quad (28)$$

WE ASSUME,

$$EV^{\wedge}_G(S)=0 \quad (29)$$

USING DC TERMS:

$$V_D - V_C = \frac{2-D}{1-D}; \qquad I_G = \frac{V_0}{RD} \qquad I_G = \frac{V_0}{RDD^1} \quad (30)$$

$$\frac{V^{\wedge}_0(S0\;)}{D(S)} = V_G \frac{-SL(2-D) + R}{\frac{(1-D)}{LRC_2 + SL + R(1-D)^2}} \quad (8)$$

IF

$C_1 = C_2 = 2 * 10^{\wedge-6}, L=10 * 10^{\wedge-3}, R=100\Omega$

$$\frac{V_0(S)}{D(S)} = \frac{S(-2.16) + 3600}{S^2(0.000002)+S(0.01)+25} \quad (9)$$

The transfer functions obtained in the state space averaging and circuit averaging technique are the same and it is shown in the Equation 3 and 8.

CHAPTER 3

MAXIMUM POWER POINT TRACKING

3. INTRODUCTION

Solar energy is the ultimate source of energy, which is naturally replenished in a short time period of time, for this reason it is called "renewable energy" or "sustainable energy". Due to the severity of the global energy crisis and environmental pollution, the photovoltaic (PV) system has become one kind of important renewable energy source. Solar energy has the advantages of Maximum reserve, inexhaustibleness, and is free from geographical restrictions, thus making PV technology a popular research topic. In this world 80 % of the green houses gases are released due to the usage of fossil fuel based. The world primary energy demand will have increased almost 60% between 2002 and 2030, averaging 1.7% increase annually, increasing still Further the greenhouse gases [5]. Oil reserves would have been exhausted by 2040, natural gas by 2060, and coal by 2300 [6]. This causes issues of high per kW installation cost but low Efficiency in PV generators. [7-8]. currently more research works has been focused on how to extract more power effectively from the PV cells. There are two ways such as solar Tracking system and maximum power point tracking (MPPT) [9, 10]. In the literature survey show that there will be an increasing30-40 % of energy will be extracted compared to the PV system without solar tracking system. The maximum power point tracking (MPPT) is usually used as online control strategy to track the maximum output power operating point of the photovoltaic generation (PVG) for different operating condition of insolation and temperature of the pvg. The author [11, 12] compares and evaluates the percentage of power extraction with MPPT and without MPPT. It clearly shows that when we use MPPT with the PV system, the power extraction efficiency is increase to 97%. The study of developing a PV charging system for li-ion batteries by integrating MPPT and charging control for the battery is reviewed.

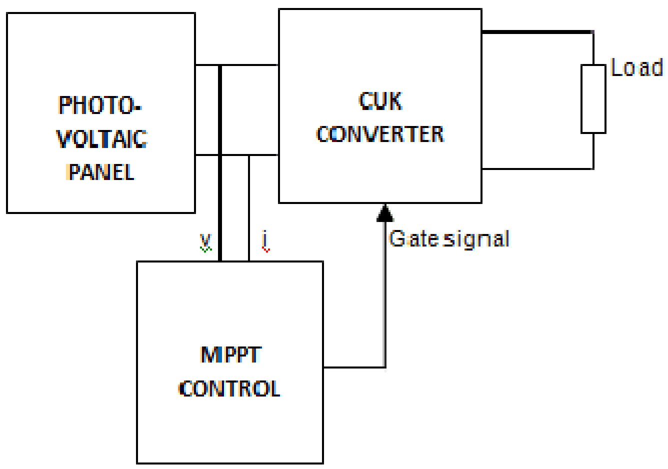


Figure 10: MPPT overall system

Photovoltaic Operation

The following figure shows a simple model of a PV cell. R_S is the series resistance associated with connecting to the active portion of a cell or module consisting of a series of equivalent cells. Using and I-V measurements, the value of R_S can be calculated. shows that R_S varies with the reciprocal of irradiance.

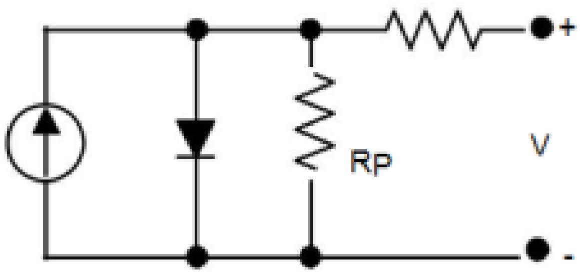


Figure 11: Model for a PV cell

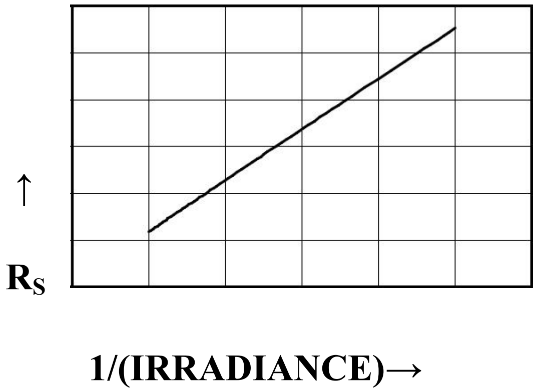


Figure 12:Rs Vs 1/(Irradiance)

Current through the diode is represented by Equation 2

$$I_O \ \text{x} \left(e^{\frac{q \ \text{x} \ (V + I \ \text{x} \ R_S)}{n \ \text{x} \ k \ \text{x} \ T}} - 1\right): \qquad (2)$$

Where:

I_O = Diode saturation current

q = Electron charge ($1.6\text{x}10^{-19}$ C)

k = Boltzmann constant ($1.38\text{x}10^{-23}$J/K)

n = Ideality factor (from 1 to 2)

T = Temperature (ºK)

The value n x k x T is weak function of ln(irradiance). This most likely is a change in the ideality factor as the irradiance changes.

The parameters usually given in PV data sheets are:

V_{OC} = Open circuit output voltage

I_{SC} = Short circuit output current

V_{MP} = Maximum power output voltage

I_{MP} = Maximum power output current

These values are typically given for 25°C and 1000W/m^2. It shows a comparison of the I-V and power characteristics at different values of irradiance.

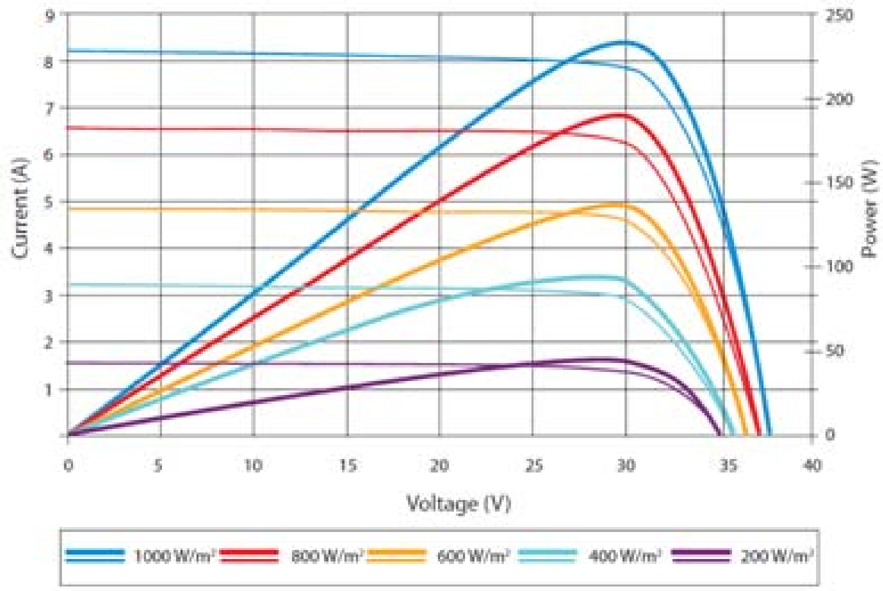


Figure 13:IV Characteristics

The I_{SC} values are proportional to the irradiance. As well, the I_{MP} changes in proportion to the irradiance as shown.

Another aspect that sometimes is overlooked is that the output current is also a function of the angle of incidence. Although the total irradiance may be constant, if the angle of incidence is not zero compared to the source, the effective irradiance is reduced which results in a reduction in current as shown. This factor may be more evident when a PV system has modules that cannot be uniformly mounted or the system is mobile. In the case where the system is mobile, the angle may be continuously changing and the maximum power point tracking system may require greater tracking speed.

3.1 MPPT Methods

One of the more complete analyses of MPPT methods is given in Reference 1. This paper compares 7 different methods along derivatives of two of the methods.

These methods include:

1. Constant Voltage
2. Open Circuit Voltage
3. Short Circuit Current
4. Perturb and Observe
5. Incremental Conductance
6. Temperature
7. Temperature Parametric

MPPT methods 1 through 5 are covered in this document.

3.1.1 Constant Voltage

The constant voltage method is the simplest method. This method simply uses single voltage to represent the V_{MP}. In some cases this value is programmed by an external resistor connected to a current source pin of the control IC. In this case, this resistor can be part of a network that includes a NTC thermistor so the value can be temperature compensated. Reference 1 gives this method an overall rating of about 80%. This means that for the various different irradiance variations, the method will collect about 80%

of the available maximum power. The actual performance will be determined by the average level of irradiance. In the cases of low levels of irradiance the results can be better.

3.1.2 Open Circuit Voltage

An improvement on this method uses V_{OC} to calculate $V_{MP.}$ Once the system obtains the V_{OC} value, V_{MP} is calculated by

$$VV_{MP} = K \times V_{oc} \quad (3)$$

3.1.3 Short Circuit Current

The short circuit current method uses a value of I_{SC} to estimate I_{MP}.

This method uses a short load pulse to generate a short circuit condition. During the short circuit pulse, the input voltage will go to zero, so the power conversion circuit must be powered from some other source. One advantage of this system is the tolerance for input capacitance compared to the V_{OC} method. The k values are typically close to 0.9 to 0.98.

3.1.4 Perturb and Observe

The Perturb and Observe algorithm is a simple technique for maximum power point tracking. It is based on controlling the duty cycle (d) of a dc-dc converter to adjust the PV array terminal voltage at the maximum power point [14]. The power output of the array is monitored every cycle and is compared to its value before each perturbation is made. If a change (either positive or negative) in the duty cycle of the dc-dc converter causes output power to increase, the duty cycle is changed in the same direction. if it causes the output power to decrease, then it is reversed to the opposite direction. The performance of the algorithm is affected by the choice of the perturbation magnitude (Δd) of the converter switching duty cycle. Large perturbations cause large output power fluctuations around the MPP while small perturbations slow down the algorithm. Modifications to this technique are published in [15], [16] and [17] to improve performance while maintaining the basic principle of operation. Illustrates the operation sequence of the algorithm.

Change in duty cycle, Δd	Effect on output power	Next perturbation, Δd (n+1)
Increase	Increase	Increase
Increase	Decrease	Decrease
Decrease	Increase	Decrease
Decrease	Decrease	Increase

Table 2:Perturbation directions for the P&O algorithm based on output power variations

Perturb and Observe (P and O) searches for the maximum power point by changing the PV voltage or current and detecting the change in PV power output. The direction of the change is reversed when the PV power decreases. P and O can have issues at low irradiance that result in oscillation. There can also be issues when there are fast changes in the irradiance which can result in initially choosing the wrong direction of search.

The designer has a choice of either changing the PV voltage or current. It shows that changes in V_{MP} are closely related to ln(irradiance) and shows that I_{MP} is proportional to irradiance. Tracking PV power by changing the PV voltage is less sensitive to changes in irradiance. This becomes more of an issue as the irradiance decreases as shown in.

So finding I_{MP} will better locate the maximum power point particularly at lower insulation.

Choosing the proper step size for the search is important. Too large will result in oscillation about the maximum power point and too small will result in slow response to changes in irradiance.

To reduce the response to noise, averaging the PV power value is important when making a direction decision. Keep in mind that whenever the system is not at the maximum power point, it is not operating at the optimal point.

3.1.5 Incremental Conductance

Incremental conductance (IC) locates the maximum power point when:

$$\frac{dI_{PV}}{dV_{PV}} + \frac{I_{PV}}{V_{PV}} = 0 \quad (1)$$

The IC uses a search technique that changes a reference or a duty cycle so that V_{PV} changes and searches for the condition of and at that condition the maximum power point has been found and searching will stop. The IC will continue to calculate dI_{PV} until the result is no longer zero. At that time, the search is started again. In some cases, a non-zero value is used for comparison so the search will not be triggered by noise.

When the left side of is greater than zero, the search will increment $V_{PV.}$ When the left side of is less than zero, the search will decrement V_{PV}.

Incremental Conductance (IC) is good for conditions of rapidly varying irradiance. However, noise may cause continuous searching so some amount of noise reduction may be needed. shows an example of the

IC method. In this case, five points were used for each test of maximum power point. This was accomplished using a least squares method to determine DVPV and I_{PV}. However, artifacts due to noise can be seen starting around 45V.

3.1.5.1 Incremental Conductance Algorithm:

This algorithm exploits the fact that the slope of the power-voltage curve of a PV array is equal to zero at the maximum power point. The slope is positive in the area to the left of the maximum power point and negative in the area to the right. Mathematically, this can be summarized as:

$$dP / dV = 0, \textit{At MPP}$$

$$dP / dV \geq 0, \text{Left of MPP}$$

$$dP / dV \leq 0, \text{Right of MPP (a)}$$

This can be simplified using the following approximation:

$$dP / dV = d\,(IV) = I + VdI / dV \approx I + V\Delta I / \Delta V \; (7)$$

From that, (a) can be rewritten as:

$$\Delta I / \Delta V = -I / V, \text{ at MMP}$$

$$\Delta I / \Delta V = -I / V, \text{ left of MMP}$$

$$\Delta I / \Delta V = -I / V, \text{ right of MMP} \qquad (8)$$

The incremental conductance algorithm is illustrated. Where Vref is used as a reference Control signal for the dc converter Similar to the P&O

algorithm, the performance of the incremental conductance MPPT is affected by the increment size of Vref, used here as the control Variable.

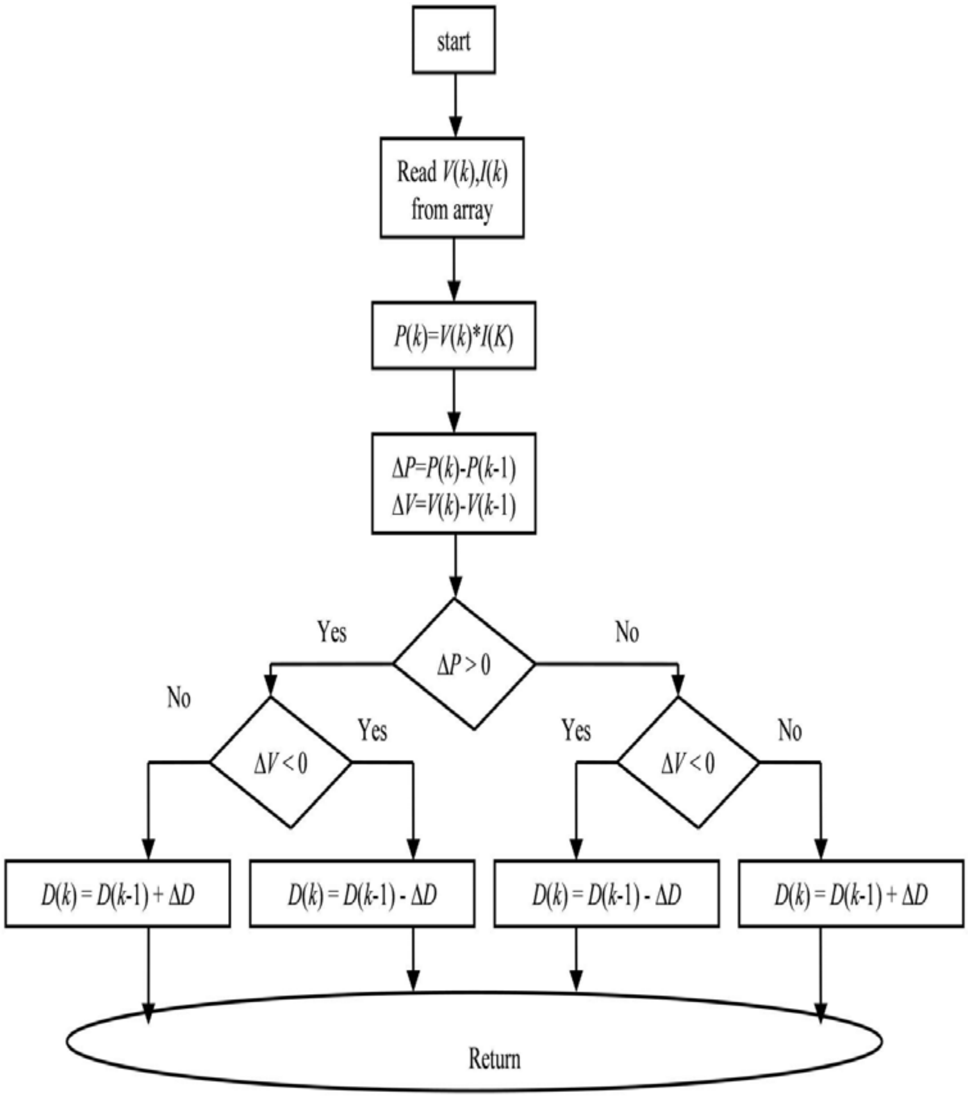


Figure 14:MPPT flowchart

If MPP lies on right side, dI/dV< -I/V and then the PV voltage must be decreased to reach the MPP [9].IC methods can be used for finding the MPP, improve the PV efficiency, reduce power loss and system cost [14]. Implementation IC on a microcontroller produced more stable performance when it compared to P&O [13]. The oscillation around MPP area also can be suppressed in trade of with its implementation complexity. Tracking time still not fast since the voltage increment and decrement had been selected manually by trial and error. IC algorithm can be seen the following figure.

3.2 BLOCK DIAGRAM OF AN MPPT WORKING MODEL:

System block diagram for a buckboost mppt converter is as shown

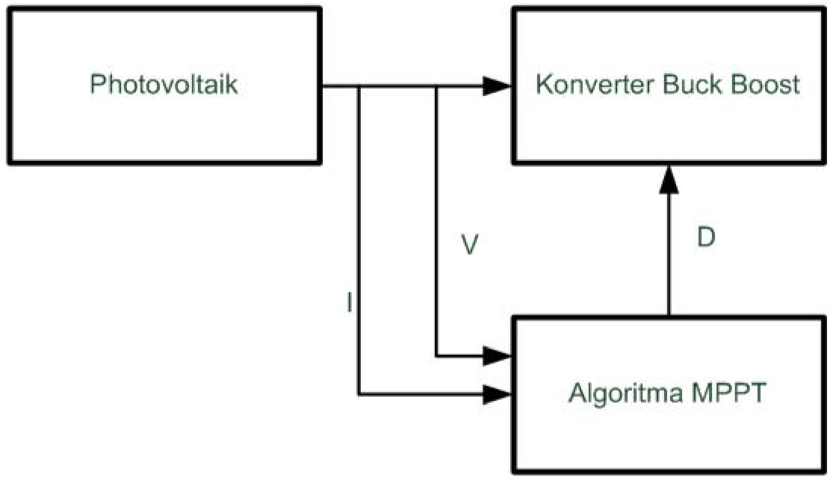


Figure 15:Block Diagram of a MPPT system

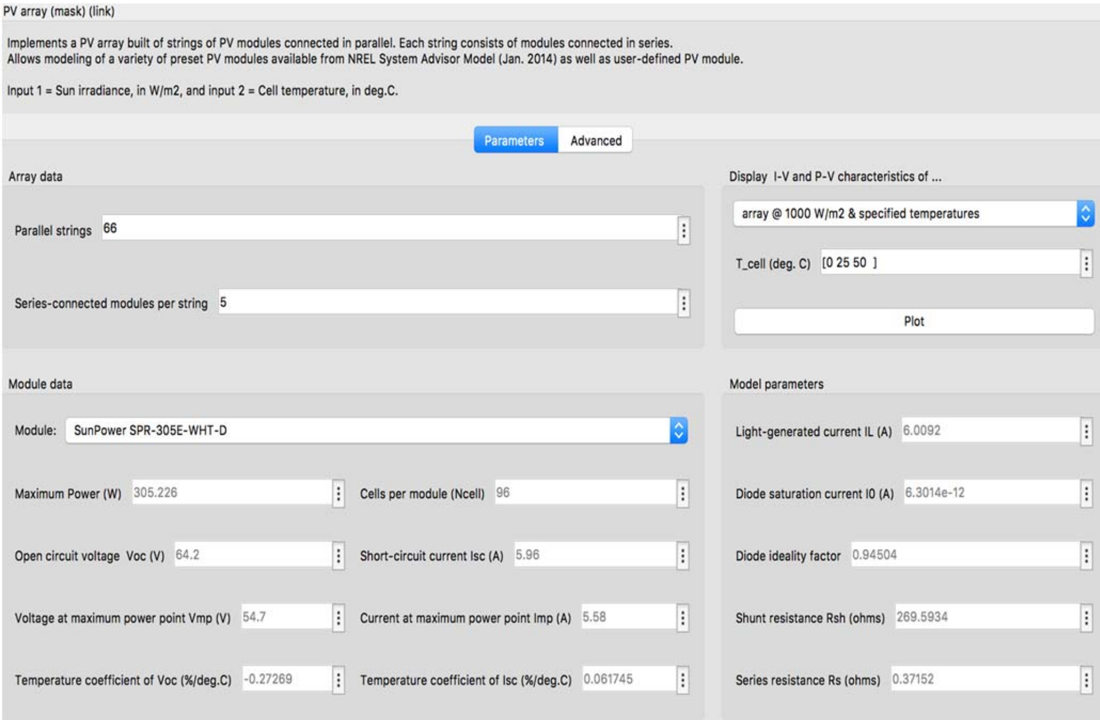
Here the mppt actually gets the current and voltage values from th pv panel and also gets the value from the output of the power converter,it compares both the values obtained,and as we know,if the value of power I the converter is less it increases the value of supply by boosting it,on the other hand if it is more it would perform the buck operation.When the power of the pv panel is equal to the converter’s output then maximum power is transferred from the source to the load.

CHAPTER 4

OVERALL SIMULATION OF PROPOSED SYSTEM:

4.1 SIMULATION RESULTS

The MATLAB/Simulink model of the Solar PV system was performed in a 100 kw Grid connected detailed model and the following figures depict the several block diagrams of the simulation involved.



4.2. SIMULATION PARAMETERS

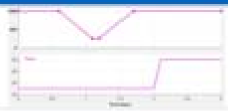
Irradiance (W/m^2)	Peaks at 1000 W/m^2
Irradiance Temperature (deg. C) Ramp-up /down	
Performance	Pmpp @ 1000 W/m^2, 25 deg= 100.7 kW @ 273.5 V Pmpp @ 250 W/m^2, 25 deg= 24.4 kW @ 265.1 V
L1, C1, C2	1e-3,1e-6,1e-6
Load resistance Rl	100 Ω

Table 4: Simulation parameters

4.3. SUPER LIFT LUO CONVERTER

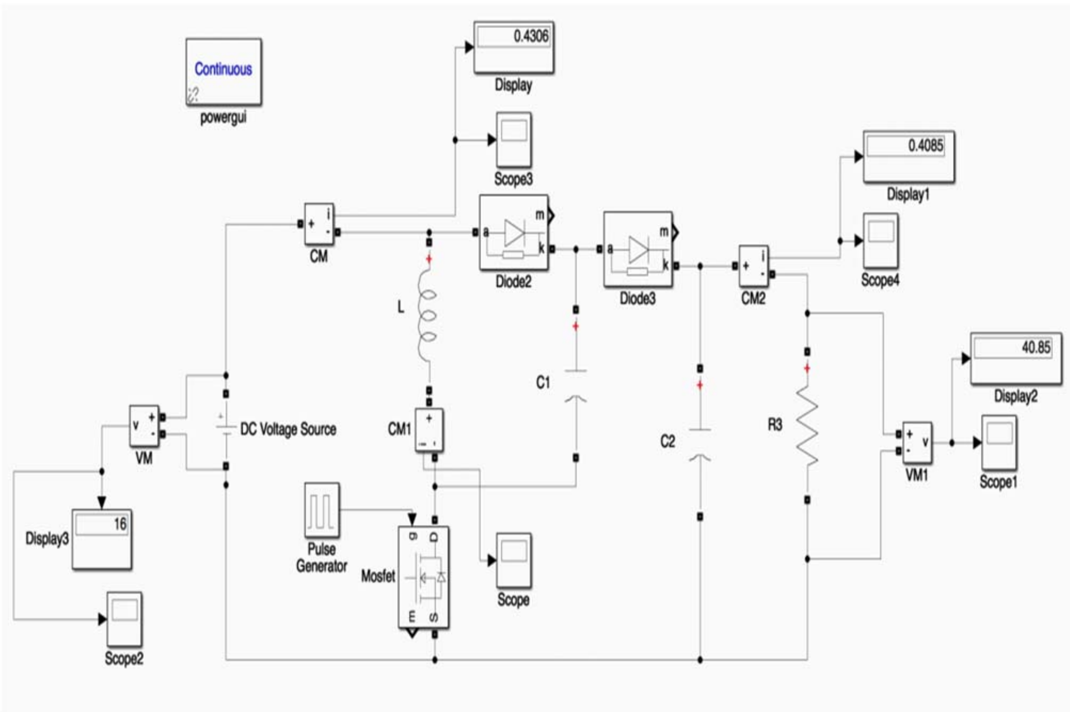


Figure 15 : Simulation of Superlift LUO Converter

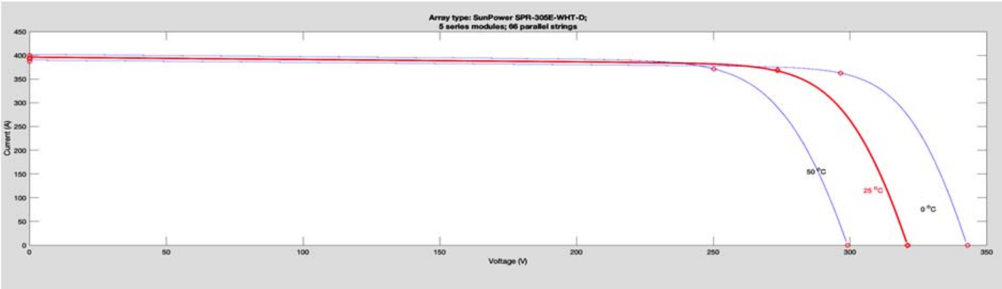


Figure 16 : Plot of PV array current vs voltage used in simulation

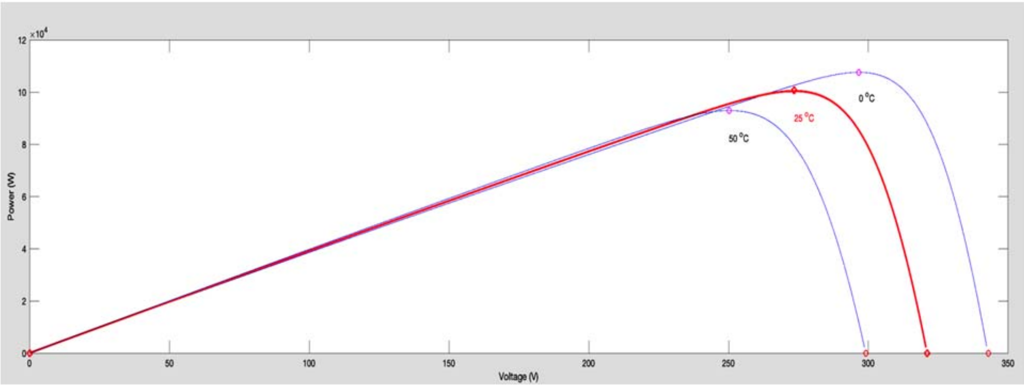


Figure 17 : Plot of PV array power vs voltage used in simulation

Results of Luo converter simulation	Values
Input voltage (V)	100V
Input current (A)	0.4306 A
Output Voltage (V)	274.4 V
Output current (A)	0.4085 A

Table 5 : Converter parameter in simulation

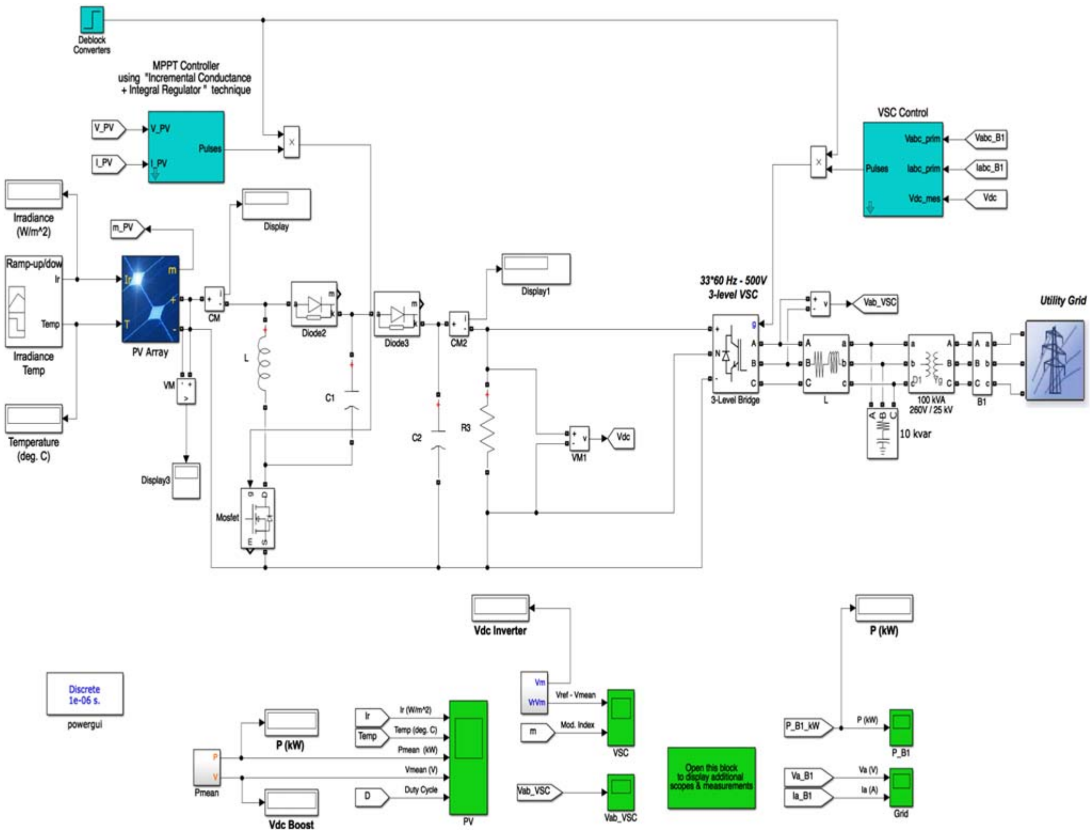


Figure 18 : Simulation of 100 KW Grid connected detailed PV System

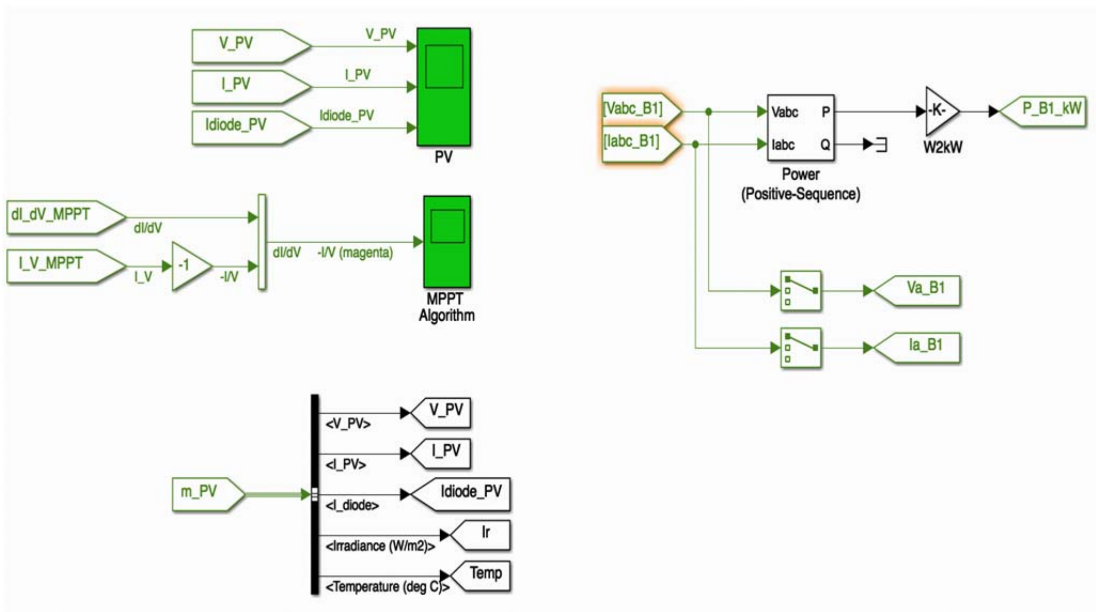


Figure 19 : MPPT simulation

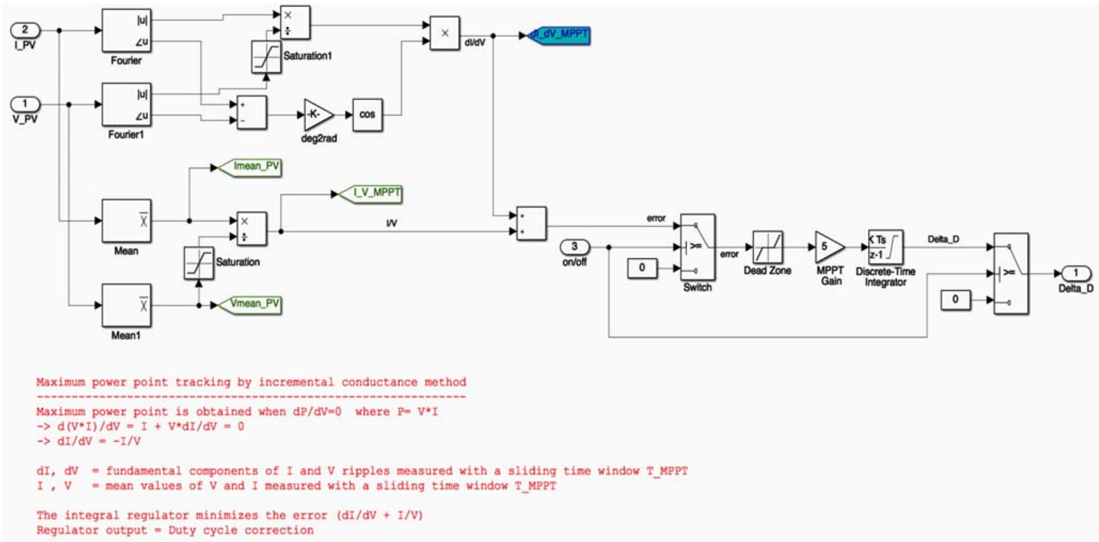


Figure 19 : MPPT by IC Algorithm system

4.4 Simulation results

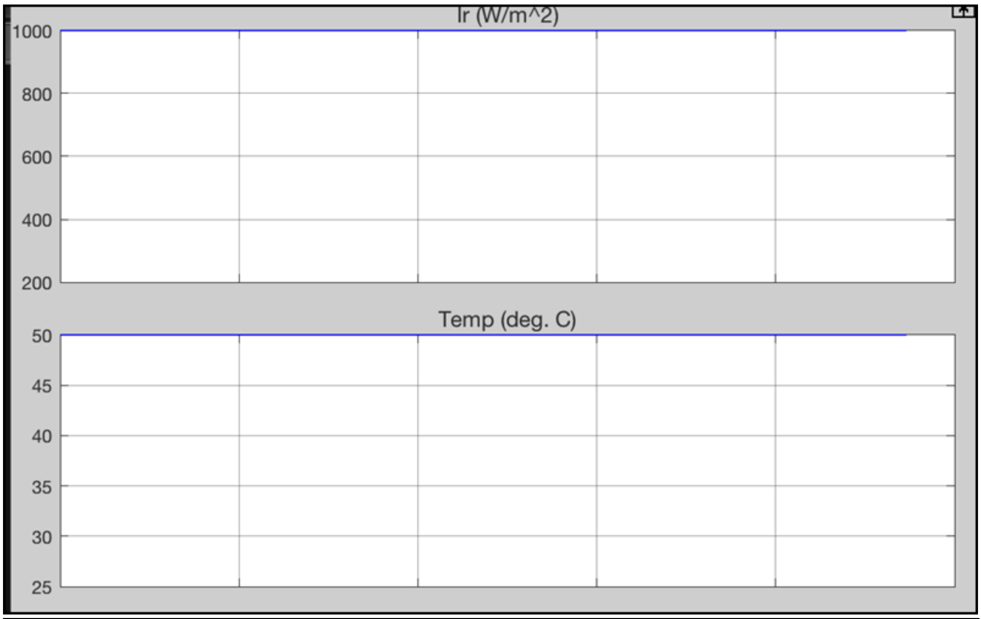


Figure 20 : Irradition and temperature Vs time

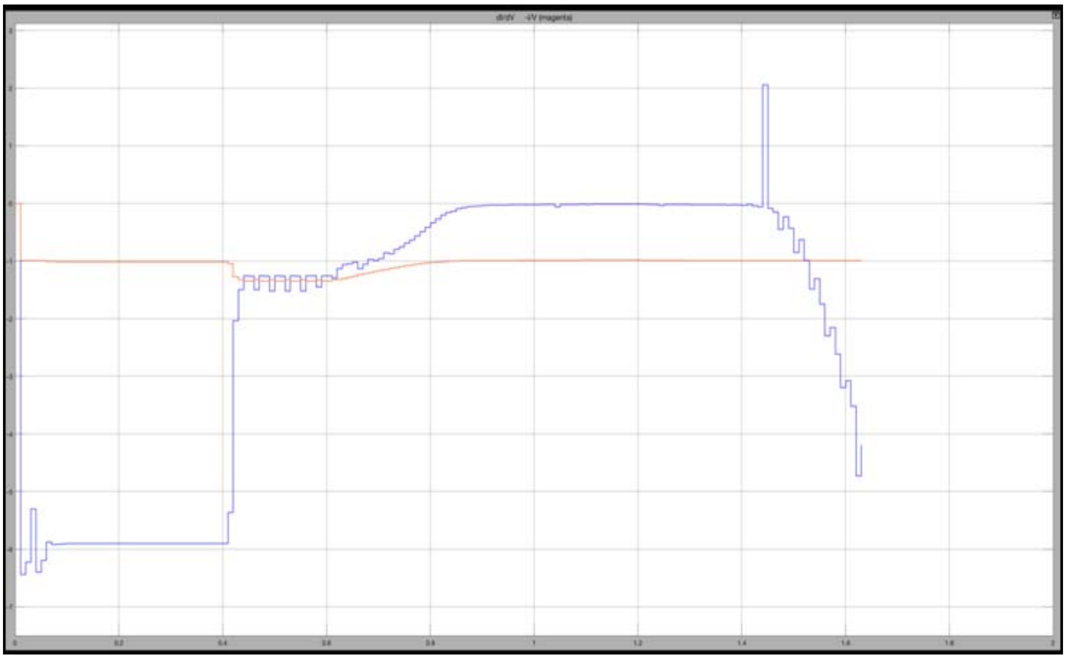


Figure 21 : Plot of Irradiance and temp in 100kw solar PV array

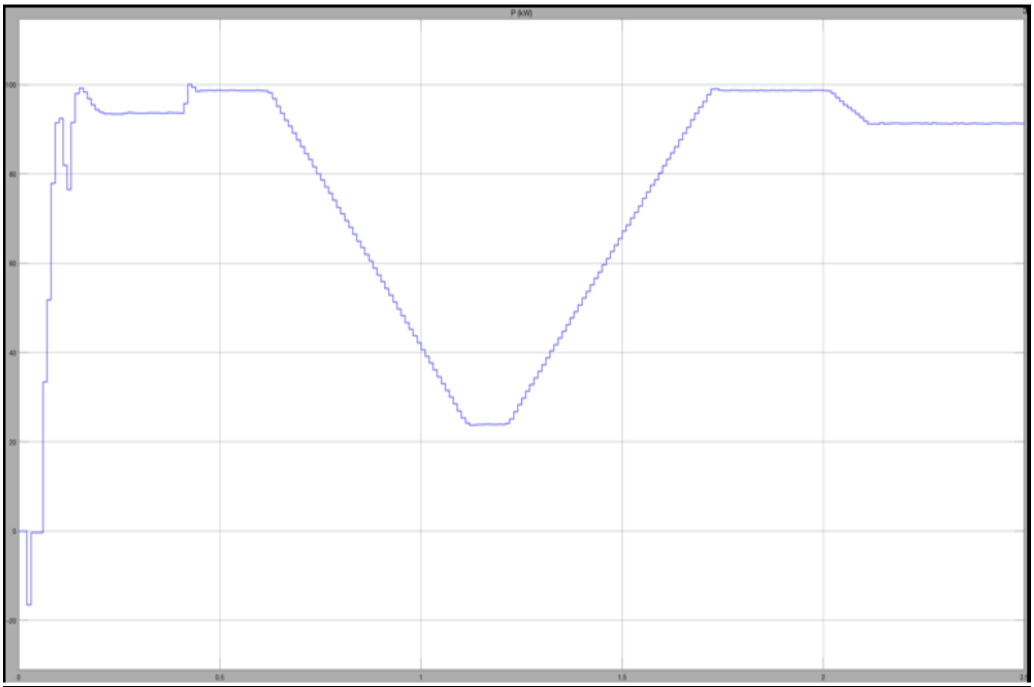


Figure 22: Power versus time

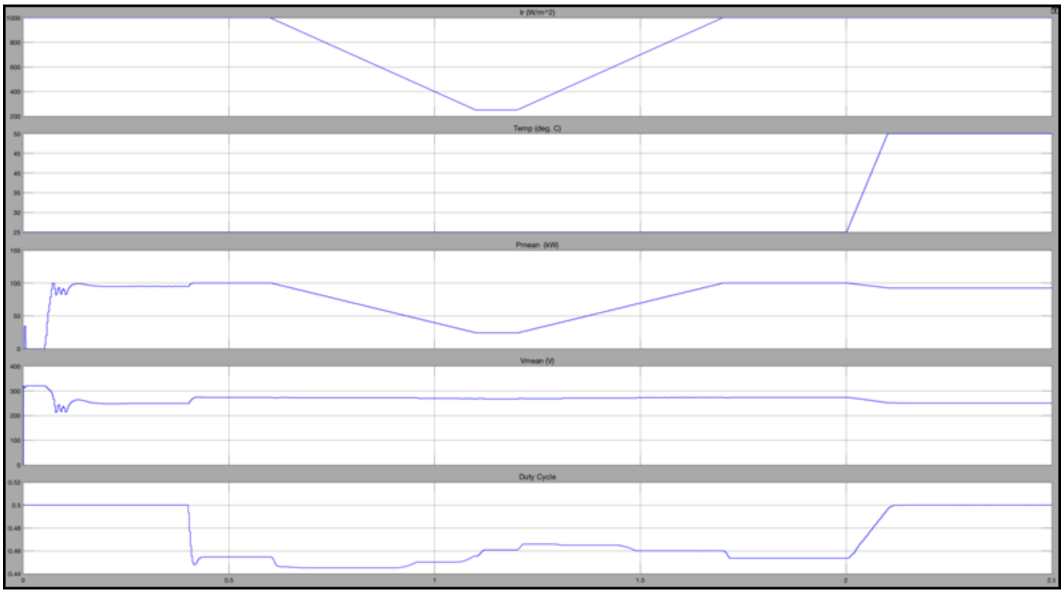


Figure 24 : Irradiation, Temperature, P_{mean}, V_{mean} Duty Cycle Vs Time

Results obtained At 1000 W/m^2 and Temp 30 deg C	**Values**
P (kW)	100.36
Vdc boost	274.44 V
Vdc Inverter	499.99

Table 4 : Simulation results from grid connected pv system

CHAPTER 5

HARDWARE IMPLEMENTATION

5.1 OVERALL SYSTEM:

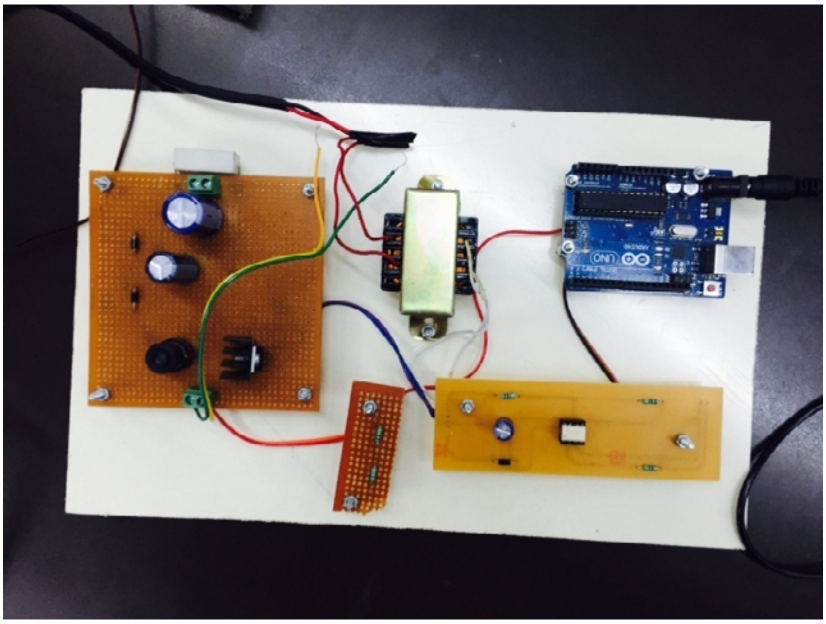


Figure 25 : Overall system

5.2 BATTERY:

Figure 26 : Battery

5.3 SOLAR PANEL:

Figure 27 : Solar panel

5.4 AURDINO:

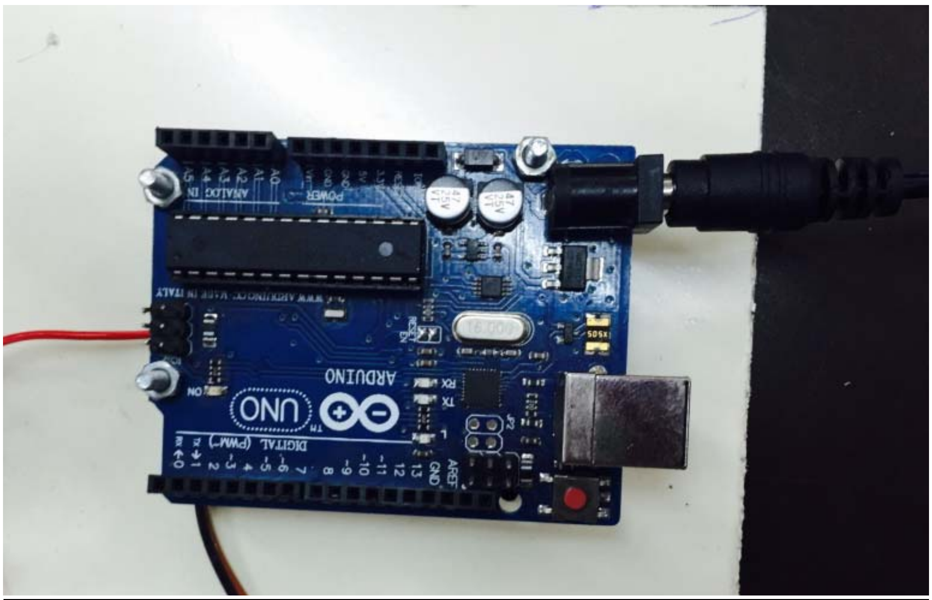


Figure 28 : Aurdino

5.4.1 Coding:

```
//  This code is for an arduino Nano based Solar MPPT charge controller.
////  Specifications :
//////////////////////////////////////////////////////////////////////////////////////////////////////////////
//
//    1.Solar panel power = 5W
//    2.Rated Battery Voltage= 12V ( lead acid type )
//    3.Maximum current = .5A
//    4.Maximum load current =10A
//    5. In put Voltage = Solar panel with Open circuit voltage from
      17 to 25V
#include "TimerOne.h"               // using Timer1 library from
http://www.arduino.cc/playground/Code/Timer1
#include <LiquidCrystal_I2C.h>      // using the LCD I2C Library from
#include <Wire.h>
//-------------------------------------------------------------------------------------------
-----------------/////// Arduino pins
Connections//////////////////////////////////////////////////////////////////////////////////
// A0 - Voltage divider (solar)
// A1 - ACS 712 Out
// A2 - Voltage divider (battery)
// A4 - LCD SDA
```

```
// A5 - LCD SCL
// D2 - ESP8266 Tx
// D3 - ESP8266 Rx through the voltage divider
// D5 - LCD back control button
// D6 - Load Control
// D8 - 2104 MOSFET driver SD
// D9 - 2104 MOSFET driver IN
// D11- Green LED
// D12- Yellow LED
// D13- Red LED
#define SOL_VOLTS_CHAN 0               // defining the adc channel to read
solar volts
#define SOL_AMPS_CHAN 1                // Defining the adc channel to read
solar amps
#define BAT_VOLTS_CHAN 2               // defining the adc channel to
read battery volts
#define AVG_NUM 8                      // number of iterations of the adc
routine to average the adc readings
// ACS 712 Current Sensor is used. Current Measured = (5/(1024
*0.185))*ADC - (2.5/0.185)
```

```
#define SOL_AMPS_SCALE  0.026393581        // the scaling value for raw adc reading to get solar amps   // 5/(1024*0.185)

#define SOL_VOLTS_SCALE 0.029296875        // the scaling value for raw adc reading to get solar volts  // (5/1024)*(R1+R2)/R2 // R1=100k and R2=20k

#define BAT_VOLTS_SCALE 0.029296875        // the scaling value for raw adc reading to get battery volts

#define PWM_PIN 9                  // the output pin for the pwm (only pin 9 avaliable for timer 1 at 50kHz)

#define PWM_ENABLE_PIN 8           // pin used to control shutoff function of the IR2104 MOSFET driver (When high, the mosfet driver is on)

#define PWM_FULL 1023              // the actual value used by the Timer1 routines for 100% pwm duty cycle

#define PWM_MAX 100                // the value for pwm duty cyle 0-100%

#define PWM_MIN 60                 // the value for pwm duty cyle 0-100% (below this value the current running in the system is = 0)

#define PWM_START 90               // the value for pwm duty cyle 0-100%

#define PWM_INC 1                  //the value the increment to the pwm value for the ppt algorithm

#define TRUE 1

#define FALSE 0

#define ON TRUE
```

#define OFF FALSE

#define TURN_ON_MOSFETS digitalWrite(PWM_ENABLE_PIN, HIGH) // enable MOSFET driver

#define TURN_OFF_MOSFETS digitalWrite(PWM_ENABLE_PIN, LOW) // disable MOSFET driver

#define ONE_SECOND 50000 //count for number of interrupt in 1 second on interrupt period of 20us

#define LOW_SOL_WATTS 5.00 //value of solar watts // this is 5.00 watts

#define MIN_SOL_WATTS 1.00 //value of solar watts // this is 1.00 watts

#define MIN_BAT_VOLTS 11.00 //value of battery voltage // this is 11.00 volts

#define MAX_BAT_VOLTS 14.10 //value of battery voltage// this is 14.10 volts

#define BATT_FLOAT 13.60 // battery voltage we want to stop charging at

#define HIGH_BAT_VOLTS 13.00 //value of battery voltage // this is 13.00 volts

#define LVD 11.5 //Low voltage disconnect setting for a 12V system

#define OFF_NUM 9 // number of iterations of off charger state

```
//----------------------------------------------------------------------------------------
//Defining led pins for indication
#define LED_RED 11
#define LED_GREEN 12
#define LED_YELLOW 13
//----------------------------------------------------------------------------------------
// Defining load control pin
#define LOAD_PIN 6       // pin-6 is used to control the load
//----------------------------------------------------------------------------------------
// Defining lcd back light pin
#define BACK_LIGHT_PIN 5       // pin-5 is used to control the lcd back
light
//----------------------------------------------------------------------------------------
//////////////////////////////////////////BIT MAP
ARRAY//////////////////////////////////////////////////
//----------------------------------------------------------------------------------------
byte solar[8] = //icon for solar panel
{
  0b11111,
  0b10101,
  0b11111,
```

```
  0b10101,
  0b11111,
  0b10101,
  0b11111,
  0b00000
};
byte battery[8]= // icon for battery
{
  0b01110,
  0b11011,
  0b10001,
  0b10001,
  0b11111,
  0b11111,
  0b11111,
  0b11111,
};
byte _PWM [8]= // icon for PWM
{
  0b11101,
```

```
  0b10101,
  0b10101,
  0b10101,
  0b10101,
  0b10101,
  0b10101,
  0b10111,
};
//-------------------------------------------------------------------------------------
// global variables
float sol_amps;                    // solar amps
float sol_volts;                  // solar volts
float bat_volts;                  // battery volts
float sol_watts;                   // solar watts
float old_sol_watts = 0;             // solar watts from previous time through
ppt routine
unsigned int seconds = 0;            // seconds from timer routine
unsigned int prev_seconds = 0;        // seconds value from previous pass
unsigned int interrupt_counter = 0;   // counter for 20us interrrupt
unsigned long time = 0;              // variable to store time the back light
control button was pressed in millis
```

int delta = PWM_INC; // variable used to modify pwm duty cycle for the ppt algorithm

int pwm = 0; // pwm duty cycle 0-100%

int back_light_pin_State = 0; // variable for storing the state of the backlight button

int load_status = 0; // variable for storing the load output state (for writing to LCD)

enum charger_mode {off, on, bulk, bat_float} charger_state; // enumerated variable that holds state for charger state machine

// set the LCD address to 0x27 for a 20 chars 4 line display

// Set the pins on the I2C chip used for LCD connections:

// addr, en,rw,rs,d4,d5,d6,d7,bl,blpol

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE); // Set the LCD I2C address

//--

// This routine is automatically called at powerup/reset

//--

void setup() // run once, when the sketch starts

{

pinMode(LED_RED, OUTPUT); // sets the digital pin as output

pinMode(LED_GREEN, OUTPUT); // sets the digital pin as output

```
  pinMode(LED_YELLOW, OUTPUT);        // sets the digital pin as
output
  pinMode(PWM_ENABLE_PIN, OUTPUT);    // sets the digital pin as
output
  Timer1.initialize(20);             // initialize timer1, and set a 20uS period
  Timer1.pwm(PWM_PIN, 0);            // setup pwm on pin 9, 0% duty
cycle
  TURN_ON_MOSFETS;                   // turn off MOSFET driver chip
  Timer1.attachInterrupt(callback);  // attaches callback() as a timer
overflow interrupt
  Serial.begin(9600);                // open the serial port at 9600 bps:
  pwm = PWM_START;                   // starting value for pwm
  charger_state = on;                // start with charger state as off
  pinMode(BACK_LIGHT_PIN, INPUT);    // backlight on button
  pinMode(LOAD_PIN,OUTPUT);          // output for the LOAD
MOSFET (LOW = on, HIGH = off)
  digitalWrite(LOAD_PIN,HIGH);       // default load state is OFF
  lcd.begin(20,4);                   // initialize the lcd for 16 chars 2 lines, turn
on backlight
  lcd.noBacklight();                 // turn off the backlight
  lcd.createChar(1,solar);           // turn the bitmap into a character
  lcd.createChar(2,battery);         // turn the bitmap into a character
```

```
  lcd.createChar(3,_PWM);              // turn the bitmap into a character
}
//------------------------------------------------------------------------------------
// Main loop
//------------------------------------------------------------------------------------
void loop()
{
  read_data();                         // read data from inputs
  run_charger();                        // run the charger state machine
  print_data();                        // print data
  load_control();                       // control the connected load
  led_output();                        // led indication
  lcd_display();                       // lcd display
}
//------------------------------------------------------------------------------------
// This routine reads and averages the analog inputs for this system, solar
volts, solar amps and
// battery volts.
//------------------------------------------------------------------------------------
int read_adc(int channel){
```

```
  int sum = 0;

  int temp;

  int i;

  for (i=0; i<AVG_NUM; i++) {          // loop through reading raw adc
values AVG_NUM number of times

    temp = analogRead(channel);       // read the input pin

    sum += temp;                      // store sum for averaging

    delayMicroseconds(50);            // pauses for 50 microseconds

  }

  return(sum / AVG_NUM);              // divide sum by AVG_NUM to get
average and return it

}

//------------------------------------------------------------------------------------

// This routine reads all the analog input values for the system. Then it
multiplies them by the scale

// factor to get actual value in volts or amps.

//------------------------------------------------------------------------------------

void read_data(void) {

  sol_amps = (read_adc(SOL_AMPS_CHAN) * SOL_AMPS_SCALE -
12.01);    //input of solar amps

  sol_volts = read_adc(SOL_VOLTS_CHAN) * SOL_VOLTS_SCALE;
//input of solar volts
```

```
  bat_volts = read_adc(BAT_VOLTS_CHAN) * BAT_VOLTS_SCALE;
//input of battery volts

  sol_watts = sol_amps * sol_volts ;                          //calculations of
solar watts

}

//------------------------------------------------------------------------------------

// This is interrupt service routine for Timer1 that occurs every 20uS.

//

//------------------------------------------------------------------------------------

void callback()

{

  if (interrupt_counter++ > ONE_SECOND) {        // increment
interrupt_counter until one second has passed

    interrupt_counter = 0;                    // reset the counter

    seconds++;                                // then increment seconds counter

  }

}

//------------------------------------------------------------------------------------

// This routine uses the Timer1.pwm function to set the pwm duty cycle.

//------------------------------------------------------------------------------------

void set_pwm_duty(void) {
```

```
  if (pwm > PWM_MAX) {                                        // check limits
of PWM duty cyle and set to PWM_MAX

    pwm = PWM_MAX;

  }

  else if (pwm < PWM_MIN) {                              // if pwm is less than
PWM_MIN then set it to PWM_MIN

    pwm = PWM_MIN;

  }

  if (pwm < PWM_MAX) {

    Timer1.pwm(PWM_PIN,(PWM_FULL * (long)pwm / 100), 20);  // use
Timer1 routine to set pwm duty cycle at 20uS period

    //Timer1.pwm(PWM_PIN,(PWM_FULL * (long)pwm / 100));

  }


  else if (pwm == PWM_MAX) {                              // if pwm set to
100% it will be on full but we have

    Timer1.pwm(PWM_PIN,(PWM_FULL - 1), 20);              // keep
switching so set duty cycle at 99.9%

    //Timer1.pwm(PWM_PIN,(PWM_FULL - 1));

  }


}
```

//--

// This routine is the charger state machine. It has four states on, off, bulk and float.

// It's called once each time through the main loop to see what state the charger should be in.

// The battery charger can be in one of the following four states:

//

// On State - this is charger state for MIN_SOL_WATTS < solar watts < LOW_SOL_WATTS. In this state isthe solar

// watts input is too low for the bulk charging state but not low enough to go into the off state.

// In this state we just set the pwm = 99.9% to get the most of low amount of power available.

// Bulk State - this is charger state for solar watts > MIN_SOL_WATTS. This is where we do the bulk of the battery

// charging and where we run the Peak Power Tracking alogorithm. In this state we try and run the maximum amount

// of current that the solar panels are generating into the battery.

// Float State - As the battery charges it's voltage rises. When it gets to the MAX_BAT_VOLTS we are done with the

// bulk battery charging and enter the battery float state. In this state we try and keep the battery voltage

// at MAX_BAT_VOLTS by adjusting the pwm value. If we get to pwm = 100% it means we can't keep the battery

// voltage at MAX_BAT_VOLTS which probably means the battery is being drawn down by some load so we need to back

// into the bulk charging mode.

// Off State - This is state that the charger enters when solar watts < MIN_SOL_WATTS. The charger goes into this

// state when there is no more power being generated by the solar panels. The MOSFETs are turned

// off in this state so that power from the battery doesn't leak back into the solar panel.

```
//------------------------------------------------------------------------------------
void run_charger(void) {

  static int off_count = OFF_NUM;

  switch (charger_state) {

    case on:

      if (sol_watts < MIN_SOL_WATTS) {                    // if watts input from the solar panel is less than

        charger_state = off;                              // the minimum solar watts then

        off_count = OFF_NUM;                              // go to the charger off state

        TURN_OFF_MOSFETS;
```

```
    }
    else if (bat_volts > (BATT_FLOAT - 0.1)) {           // else if the battery
voltage has gotten above the float
      charger_state = bat_float;                        // battery float voltage go to
the charger battery float state
    }
    else if (sol_watts < LOW_SOL_WATTS) {               // else if the solar
input watts is less than low solar watts
      pwm = PWM_MAX;                                    // it means there is not
much power being generated by the solar panel
      set_pwm_duty();                                   // so we just set the pwm
= 100% so we can get as much of this power as possible
    }                                                   // and stay in the charger on state
    else {
      pwm = ((bat_volts * 10) / (sol_volts / 10)) + 5;    // else if we are
making more power than low solar watts figure out what the pwm
      charger_state = bulk;                             // value should be and change
the charger to bulk state
    }
    break;
  case bulk:
```

```
    if (sol_watts < MIN_SOL_WATTS) {                    // if watts input
from the solar panel is less than
      charger_state = off;                              // the minimum solar watts
then it is getting dark so
      off_count = OFF_NUM;                              // go to the charger off
state
      TURN_OFF_MOSFETS;
    }
    else if (bat_volts > BATT_FLOAT) {                  // else if the battery
voltage has gotten above the float
      charger_state = bat_float;                        // battery float voltage go to
the charger battery float state
    }
    else if (sol_watts < LOW_SOL_WATTS) {               // else if the solar
input watts is less than low solar watts
      charger_state = on;                               // it means there is not much
power being generated by the solar panel
      TURN_ON_MOSFETS;                                  // so go to charger on
state
    }
    else {                                              // this is where we do the Peak
Power Tracking ro Maximum Power Point algorithm
```

```
        if (old_sol_watts >= sol_watts) {                  // if previous watts are
greater change the value of
          delta = -delta;                                    // delta to make pwm
increase or decrease to maximize watts
        }
        pwm += delta;                                    // add delta to change PWM
duty cycle for PPT algorythm (compound addition)
        old_sol_watts = sol_watts;                       // load old_watts with
current watts value for next time
        set_pwm_duty();                                      // set pwm duty cycle
to pwm value
      }
      break;
    case bat_float:
      if (sol_watts < MIN_SOL_WATTS) {                    // if watts input
from the solar panel is less than
        charger_state = off;                             // the minimum solar watts
then it is getting dark so
        off_count = OFF_NUM;                               // go to the charger off
state
        TURN_OFF_MOSFETS;
        set_pwm_duty();
```

```
    }
    else if (bat_volts > BATT_FLOAT) {                    // If we've charged
the battery abovethe float voltage
      TURN_OFF_MOSFETS;                                   // turn off MOSFETs
instead of modiflying duty cycle
      pwm = PWM_MAX;                                      // the charger is less
efficient at 99% duty cycle
      set_pwm_duty();                                 // write the PWM
    }
    else if (bat_volts < BATT_FLOAT) {                    // else if the battery
voltage is less than the float voltage - 0.1
      pwm = PWM_MAX;
      set_pwm_duty();                                     // start charging again
      TURN_ON_MOSFETS;
      if (bat_volts < (BATT_FLOAT - 0.1)) {               // if the voltage drops
because of added load,
      charger_state = bulk;                           // switch back into bulk state
to keep the voltage up
      }
    }
    break;
```

```
  case off:                                    // when we jump into the charger
off state, off_count is set with OFF_NUM
    TURN_OFF_MOSFETS;
    if (off_count > 0) {                       // this means that we run
through the off state OFF_NUM of times with out doing
      off_count--;                             // anything, this is to allow the
battery voltage to settle down to see if the
    }                                          // battery has been disconnected
    else if ((bat_volts > BATT_FLOAT) && (sol_volts > bat_volts)) {
      charger_state = bat_float;               // if battery voltage is still
high and solar volts are high
    }
    else if ((bat_volts > MIN_BAT_VOLTS) && (bat_volts <
BATT_FLOAT) && (sol_volts > bat_volts)) {
      charger_state = bulk;
    }
    break;
  default:
    TURN_OFF_MOSFETS;
    break;
 }
}
```

```
//----------------------------------------------------------------------------------------
////////////////////////////////////////////LOAD
CONTROL/////////////////////////////////////////////////////
//----------------------------------------------------------------------------------------
void load_control(){
  if ((sol_watts < MIN_SOL_WATTS) && (bat_volts > LVD)){   // If the
panel isn't producing, it's probably night
    digitalWrite(LOAD_PIN, LOW);                    // turn the load on
    load_status = 1;                              // record that the load is on
  }
  else{                                  // If the panel is producing, it's day
time
    digitalWrite(LOAD_PIN, HIGH);                   // turn the load off
    load_status = 0;                              // record that the load is off
  }
}
//----------------------------------------------------------------------------------------
// This routine prints all the data out to the serial port.
//----------------------------------------------------------------------------------------
void print_data(void) {
  Serial.print(seconds,DEC);
  Serial.print("");
```

```
Serial.print("Charging = ");
if (charger_state == on) Serial.print("on   ");
else if (charger_state == off) Serial.print("off  ");
else if (charger_state == bulk) Serial.print("bulk ");
else if (charger_state == bat_float) Serial.print("float");
Serial.print("");
Serial.print("pwm = ");
Serial.print(pwm,DEC);
Serial.print("");
Serial.print("Current (panel) = ");
Serial.print(sol_amps);
Serial.print("");
Serial.print("Voltage (panel) = ");
Serial.print(sol_volts);
Serial.print("");
Serial.print("Power (panel) = ");
Serial.print(sol_volts);
Serial.print("");
Serial.print("Battery Voltage = ");
Serial.print(bat_volts);
```

```
  Serial.print("");
  Serial.print("\n\r");
  //delay(1000);
}
//------------------------------------------------------------------------------------------
//----------------------------------Led Indication-------------------------------------
//------------------------------------------------------------------------------------------
void led_output(void)
{
  if(bat_volts > 14.1 )
  {
     leds_off_all();
     digitalWrite(LED_YELLOW, HIGH);
  }
  else if(bat_volts > 11.9 && bat_volts < 14.1)
  {
     leds_off_all();
     digitalWrite(LED_GREEN, HIGH);
  }
  else if(bat_volts < 11.8)
  {
```

```
      leds_off_all;
      digitalWrite(LED_RED, HIGH);
  }
}
//------------------------------------------------------------------------------------------
//
// This function is used to turn all the leds off
//
//------------------------------------------------------------------------------------------
void leds_off_all(void)
{
  digitalWrite(LED_GREEN, LOW);
  digitalWrite(LED_RED, LOW);
  digitalWrite(LED_YELLOW, LOW);
}
//------------------------------------------------------------------------------------------
//------------------------- LCD DISPLAY -----------------------------------------
//------------------------------------------------------------------------------------------
void lcd_display()
{
  back_light_pin_State = digitalRead(BACK_LIGHT_PIN);
  if (back_light_pin_State == HIGH)
```

```
 {
   time = millis();                    // If any of the buttons are pressed, save
the time in millis to "time"
 }
lcd.setCursor(0, 0);
 lcd.print("SOL");
 lcd.setCursor(4, 0);
 lcd.write(1);
 lcd.setCursor(0, 1);
 lcd.print(sol_volts);
 lcd.print("V");
 lcd.setCursor(0, 2);
 lcd.print(sol_amps);
 lcd.print("A");
 lcd.setCursor(0, 3);
 lcd.print(sol_watts);
 lcd.print("W ");
 lcd.setCursor(8, 0);
 lcd.print("BAT");
 lcd.setCursor(12, 0);
```

```
lcd.write(2);
lcd.setCursor(8, 1);
lcd.print(bat_volts);
lcd.setCursor(8,2);
if (charger_state == on)
{
lcd.print("");
lcd.setCursor(8,2);
lcd.print("on");
}
else if (charger_state == off)
{
lcd.print("");
lcd.setCursor(8,2);
lcd.print("off");
}
else if (charger_state == bulk)
{
lcd.print("");
lcd.setCursor(8,2);
```

```
lcd.print("bulk");
}
else if (charger_state == bat_float)
{
lcd.print("");
lcd.setCursor(8,2);
lcd.print("float");
}
//----------------------------------------------------------
//--------------------Battery State Of Charge ---------------
//----------------------------------------------------------
lcd.setCursor(8,3);
if ( bat_volts >= 12.7)
lcd.print( "100%");
else if (bat_volts >= 12.5 && bat_volts < 12.7)
lcd.print( "90%");
else if (bat_volts >= 12.42 && bat_volts < 12.5)
lcd.print( "80%");
else if (bat_volts >= 12.32 && bat_volts < 12.42)
lcd.print( "70%");
```

```
else if (bat_volts >= 12.2 && bat_volts < 12.32)
lcd.print( "60%");
else if (bat_volts >= 12.06 && bat_volts < 12.2)
lcd.print( "50%");
else if (bat_volts >= 11.90 && bat_volts < 12.06)
lcd.print( "40%");
else if (bat_volts >= 11.75 && bat_volts < 11.90)
lcd.print( "30%");
else if (bat_volts >= 11.58 && bat_volts < 11.75)
lcd.print( "20%");
else if (bat_volts >= 11.31 && bat_volts < 11.58)
lcd.print( "10%");
else if (bat_volts < 11.3)
lcd.print( "0%");
//------------------------------------------------------------------------
//------------------Duty Cycle----------------------------------------
//------------------------------------------------------------------------
lcd.setCursor(15,0);
lcd.print("PWM");
lcd.setCursor(19,0);
```

```
lcd.write(3);
lcd.setCursor(15,1);
lcd.print("");
lcd.setCursor(15,1);
lcd.print(pwm);
lcd.print("%");
//--------------------------------------------------------------------
//------------------------Load Status----------------------------------
//--------------------------------------------------------------------
lcd.setCursor(15,2);
lcd.print("Load");
lcd.setCursor(15,3);
if (load_status == 1)
{
 lcd.print("");
  lcd.setCursor(15,3);
  lcd.print("On");
}
else
{
```

```
  lcd.print("");
  lcd.setCursor(15,3);
  lcd.print("Off");
 }
 backLight_timer();                    // call the backlight timer function in
every loop
}
void backLight_timer(){
  if((millis() - time) <= 15000)         // if it's been less than the 15 secs, turn
the backlight on
      lcd.backlight();                  // finish with backlight on
  else
      lcd.noBacklight();                // if it's been more than 15 secs, turn the
backlight off
}
```

5.5 DRIVER CIRCUIT:

```
// Code for dc voltage measurement by using a voltage divider circuit
   int temp=0;
  float sum =0;
  float VOLTS_SCALE =0;
  float volt=0;
  void setup()
```

```
{
Serial.begin(9600);
}
void loop()
{
  for(int i = 0; i < 100; i++)  // loop through reading raw adc values 100 number of times
{
  temp=analogRead(A0);     // read the input pin
  sum += temp;             // store sum for averaging
  delay(2);
}
 sum=sum/100;              // divide sum by 100 to get average
// Calibration  for Voltage
  VOLTS_SCALE = 0.00488 * (120/20); // The voltage divider resistors are R1=100k and R2=20k // 5/1024 =0.00488
 volt = VOLTS_SCALE * sum ;
  Serial.print(volt);
Serial.println("V");
delay(500);
}
```

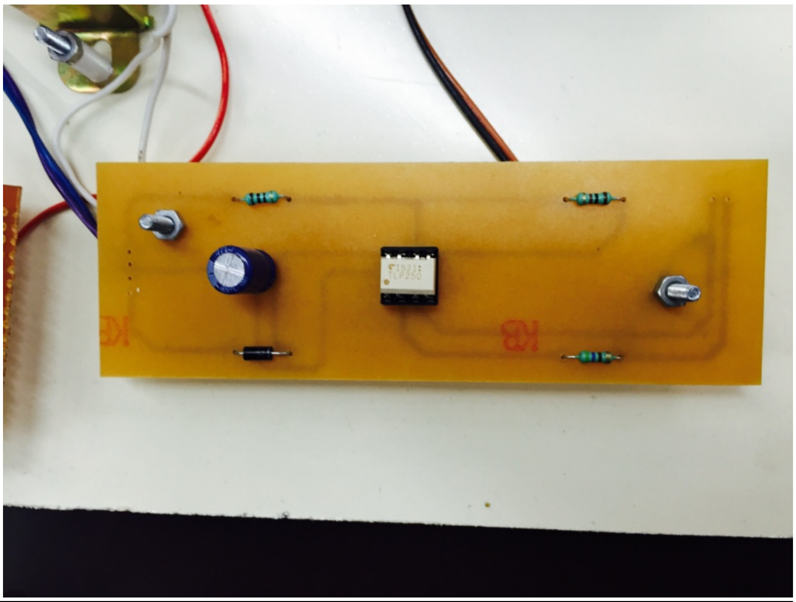


Figure 29 : Driver Circuit

5.6 VOLTAGE DIVIDER:

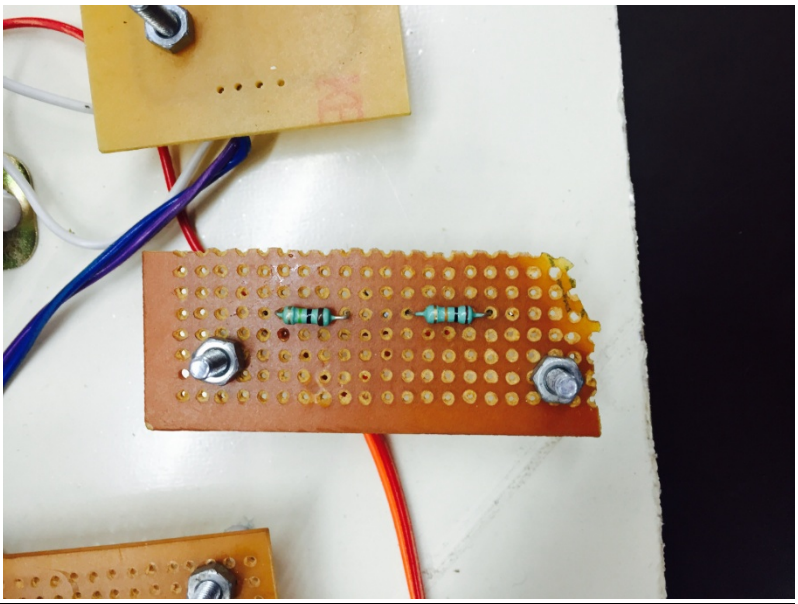


Figure 30 : Voltage Divider

5.7 SUPERLIFT LUO CONVERTER:

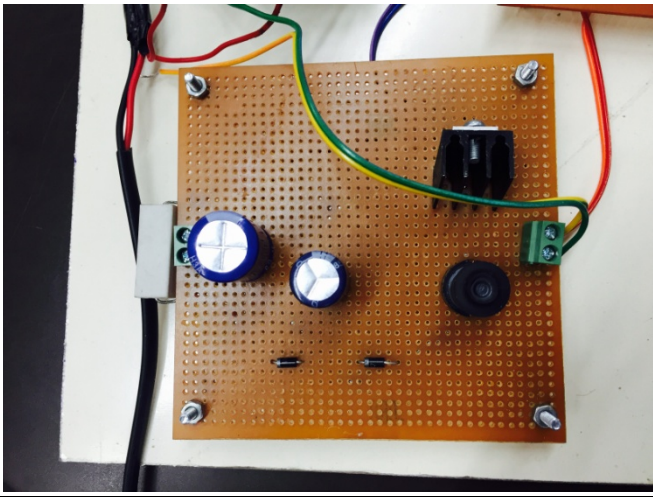


Figure 31 : Superlift Luo Converter

```
//converter test code
#include <TimerOne.h>
void setup()
{
 // Initialize the digital pin as an output.
 // Pin 13 has an LED connected on most Arduino boards
 pinMode(13, OUTPUT);
 pinMode(9, OUTPUT);
 pinMode(8, OUTPUT);
 digitalWrite(8, HIGH);
 Timer1.initialize(20); // set a timer of length 8uS
 //Timer1.attachInterrupt( timerIsr ); // attach the service routine here
 //Set duty cycle
 //Timer1.pwm(9,255.75); // 25% duty cycle
// Timer1.pwm(9, 512); //50% duty cycle
```

```
  Timer1.pwm(9, 767.25); // 75% duty cycle
}
void loop()
{
  // Main code loop
  // TODO: Put your regular (non-ISR) logic here
}
 /// --------------------------
/// Custom ISR Timer Routine
/// --------------------------
void timerIsr()
{
    // Toggle LED
    //digitalWrite( 13, digitalRead( 13 ) ^ 1 );
}
```

CHAPTER 6

HARDWARE RESULTS

Hardware parameters used in this project is a polycrystalline solar panel with 5W maximum power output. The below table lists out the various parameters of the panel

Table 5: Hardware parameters of PV module

PV Module	Rating
Pmax	5 W
Vmp	6V
Imp	0.61 A
Isc	0.7 A
Tolerance	+/- 2%

Table 6: Results obtained from different power converter topologies

<table>
<tr><th>Comparison of different DC-DC converters
At different insolation levels the output was observed</th><th>Results</th></tr>
<tr><td>Buck, Boost, Buck-Boost Converters</td><td><table>
<tr><th>Load</th><th>Without converter (%)</th><th>Buck converter (%)</th><th>Boost converter (%)</th><th>Buck-boost converter (%)</th></tr>
<tr><td>$R_L = 5\Omega$</td><td>88.5</td><td>97.2</td><td>91.2</td><td>99.9</td></tr>
<tr><td>$R_L = 20\Omega$</td><td>40.2</td><td>40.3</td><td>99.7</td><td>99.9</td></tr>
</table></td></tr>
<tr><td>Super lift Luo Converter : Simulated result with 100 kW panel</td><td>P (kW) = 100.36, Vdc boost = 274.44 V at 1000 W/m2 insolation</td></tr>
<tr><td>Super lift Luo converter : Hardware results</td><td>P (W) = 5 W, Vmp = 85 V, Imp = 0.61 A</td></tr>
</table>

CONCLUSION

The output voltage of the solar panel depends upon the insolation level and temperature and hence it varies during the day. It can be maintained constant with the help of proposed Luo converter by properly controlling the duty cycle of the MOSFETs. The power obtained from the panel also varies with the insolation level and during early morning and late evening hours; the power obtained is very less. Applying the concept of MPPT and properly controlling the duty cycle of the MOSFETs, the solar powered Luo converter successfully worked to extract maximum power from the panel.

APPENDIX 1

DRIVER UNIT:

TLP250 is suitable for gate driving circuit of IGBT or power MOS FET.

PIN CONFIGURATION:

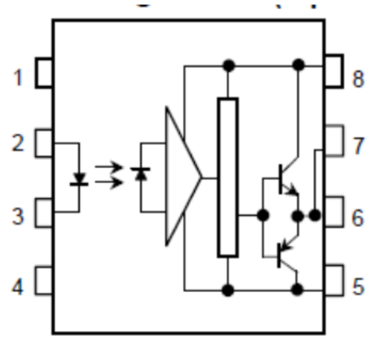


Figure 28:TLP250

Pin 1: N.C.

Pin 2: Anode

Pin 3: Cathode

Pin 4: N.C.

Pin 5: GND

Pin 6: VO (Output)

Pin 7: VO

Pin 8: VCC

Table 7:Truth Table

		Tr1	Tr2
Input LED	On	On	Off
	Off	Off	On

OPERATION WITH PIN DETAILS:

Pin 8 is VCC – the positive supply. Pin 5 is GND – the ground supply or the return path for the driving power supply. The supply voltage must be at least 10V. The maximum voltage is dependent on the operating temperature. If the temperature is lower than 70°C, up to 30V can be used. For temperatures between 70°C and 85°C, up to 20V can be used. However, there shouldn't be a need to use higher than 20V anyways. In most cases, you'll be using 12V or 15V or perhaps in some cases 18V.

Pins 2 and 3 are the inputs to the LED, anode and cathode respectively. Like regular LEDs, it has an input forward voltage and a peak forward current. The forward voltage will typically be between 1.6V and 1.8V. The forward current should be less than 20mA. The threshold input current for output transition from low to high is typically 1.2mA, but may be as high as 5mA. Thus, 10mA current should be good.

Even though pins 6 and 7 are shown to be internally connected, the output should be taken from pin 6 as the image - datasheet - shows pin 6 labeled as Vo (Output). Output voltage will tend to rise to supply voltage when high (it will actually be slightly lower) and fall to ground level when low.

The TLP250, being an optically isolated driver, has relatively slow propagation delays (not to say that optically isolated drivers can't be fast; there are optically isolated drivers faster than TLP250). The propagation delay time will typically lie between 0.15µs and 0.5µs. An important thing to remember is that the datasheet specifies the maximum operating frequency to be 25 kHz. I've used the TLP250 for frequencies up to about 16 kHz.

That covers the different parameters related to TLP250. Now let's go to the design stage and look at a few circuits. One thing you MUST remember to do when designing circuits with TLP250 is that, a 0.1μF bypass capacitor (ceramic capacitor) should be connected between V+ (pin 8) and V- (pin 5). This capacitor stabilizes the operation of the high gain linear amplifier in the TLP250. Failure to provide this capacitor may impair the switching property. The capacitor should be placed as close to the TLP250 as possible. The closer, the better.

APPENDIX II

ARUDUINO

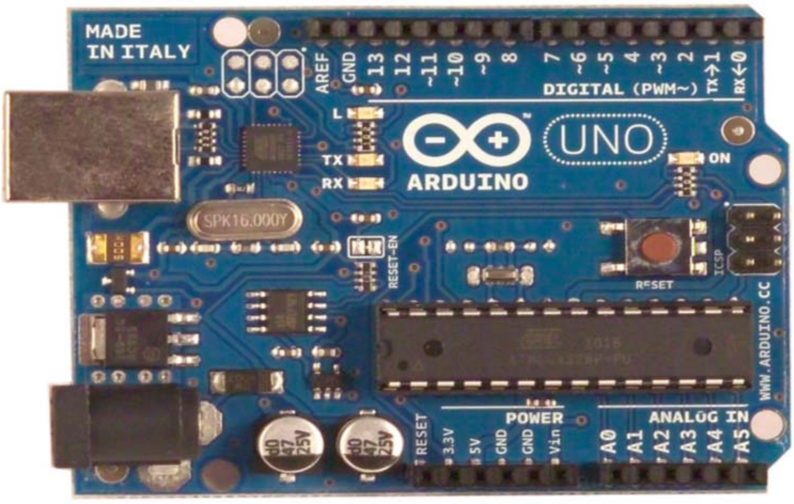


Figure 29: Aurdino

The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital Input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to Support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. The Uno and version 1.0will be the reference versions of Arduno, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions

REFERENCES

[1] Alexis Kwasinski, "Identification of Feasible Topologies for Multiple-Input DC–DC Converters, " in IEEE Transactions on power electronics, vol. 24, no. 3, march 2009.

[2] Alireza Khaligh, , Jian Cao, and Young-Joo Lee,, " A Multiple-Input DC–DC Converter Topology ," in IEEE Transactions on power electronics, vol. 24, no. 3, march 2009.

[3] Farzam Nejabatkhah, Saeed Danyali, Seyed Hossein Hosseini, Mehran Sabahi and Seyedabdolkhalegh Mozaffari Niapour, "Modeling and Control of a New Three-Input DC–DC Boost Converter for Hybrid PV/ FC/Battery Power System," in IEEE Transactions on power electronics, vol. 27, no. 5, May 2012.

[4] Fang Lin Luo, and Hong Ye, "Positive Output Super-Lift Converters," in IEEE Transactions on power electronics, vol. 18, no. 5, September 2003.

[5] F.L. Luo, "Efficiency Positive output Luo converters: voltage lift technique," in IEE Proc-Electr. Power Appl., Vol. 146, No. 2, March 1999.

[6] B. Mangu, K. Kiran Kumar and B. G. Fernandes, "Efficiency Improvement of Solar-Wind based Dual-Input Converter for Telecom Power Supply," in Proc. IEEE International Conference on Environment and Electrical Engineering. (EEEIC). pp. 914-919, May. 2012.

[7] Mohammad Taufik, Taufik Taufik, Taffy Wong, "Multiple-Input Single-Output Converter for Renewable Energy Sources," in 2012 IEEE Symposium on Industrial Electronics and Applications (ISIEA2012), September 23-26 2012 Bandug Indonesia.

[8] R. Noroozian, M. Abedi, G. B. Gharehpetian and S. H. Hosseini, "On-grid and Off-grid Operation of Multi-Input Single-Output DC/DC Converter based Fuel Cell Generation System," in Iranian Journal of Electrical & Electronic Engineering, Vol.

[9] Sobuj Kumar Ray, Diponkar Paul, Tabassum E Nur, and Kamal Chandra Paul, " A Preview on Simulation of Super-Lift Converter for Grid Connected Solar Installation," in International Journal of Innovation, Management and Technology, Vol. 3, No. 2, April 2012.

[10] Saeed Danyali, Seyed Hossein Hosseini,and Gevorg B. Gharehpetian, "New Extendable Single-Stage Multi-input DC–DC/AC Boost Converter," in IEEE Transactions on power electronics, vol. 29, no. 2, February 2014.

[11] V. Usharani & R. Agasthiya, " Enhanced MPPT Technique For DC-DC Luo Converter Using Model Predictive Control For Photovoltaic Systems," in International Journal of Engineering Research and Development, Volume 11, Issue 01 (January 2015).

[12] Zeng-li Shan, Shuo Llu, Fang-lin Luo, " Investigation of a super-lift luo-converter used in solar panel system," in International Conference on Electricity Distribution (CICED 2012) Shanghai, 5-6 Sep. 20 I 2, China.