The Islamic University – Gaza
Deanery of Graduate Studies
Faculty of Engineering
Electrical Engineering

الجامعة الإسلامية – غزة
عمادة الدراسات العليا
كلية الهندســـــــة
الهندسة الكهربائية

Sun and Maximum Power Point Tracking in Solar Array Systems Using Fuzzy Controllers Via FPGA

Mohammed S. EL-Moghany

Advisor
Dr. Basil Hamed

This Thesis is submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in
Electrical Engineering

1432هـ – 2011 م

Dedication

To my parents, big brother, sisters, wife, and lovely kids

Mohammed Saker Elmoghany

Acknowledgments

At the beginning, I thank ALLAH for giving me the strength and health to let this work see the light.

I wish to express my deepest gratitude to my advisor, Dr. Basil Hamed, for his professional assistance, support, advice and guidance throughout my thesis, and to my discussion committee, Dr. Hatem El Aydi and Dr. Assad Abu Jasser for their acceptance to discuss my thesis.

I would also like to extend my gratitude to my family for providing all the preconditions necessary to complete my studies.

Abstract

Sun and Maximum Power Point Tracking in Solar Array Systems Using Fuzzy Controllers Via FPGA

By

Mohammed S. EL-Moghany

Solar energy is viewed as clean and renewable source of energy for the future. So the use of Photovoltaic systems has increased in many applications. That need to improve the materials and methods used to harness this power source. There are two major approaches; sun tracking and maximum power point tracking. These two methods need efficient controllers. The controller may be conventional or intelligent such as Fuzzy Logic Controller (FLC). FLCs have the advantage to be relatively simple to design as they do not require the knowledge of the exact model and work well for nonlinear system. To implement this controller, Field Programmable Gate Array (FPGA) can be used. This method has many advantages over classical microprocessors.

In this research, Two fuzzy logic controllers are fabricated on modern FPGA card (Spartan-3AN, Xilinx Company, 2009) to increase the energy generation efficiency of solar cells. These controllers are, sun tracking controller and maximum power point tracking controller.

Sun tracking generating power system is designed and implemented in real time. A tracking mechanism composed of photovoltaic module, stepper motor, sensors, input/output interface and expert FLC controller implemented on FPGA, that to track the sun and keep the solar cells

always face the sun in most of the day time. The proposed sun tracking controller is tested using Matlab/Simulink program, the results show that the controller have a good response.

Maximum power point tracking system is designed and implemented. The system composed of photovoltaic module, buck converter and the fuzzy logic controller implemented on FPGA for controlling on/off time of MOSFET switch of a buck converter. The proposed maximum power point tracking controller for grid-connected photovoltaic system is tested using model designed by Matlab/Simulink program with graphical user interface (GUI) for entering the parameters of any array model using information from its datasheet, the results show that the controller have a response better than conventional controller applied on the same system.

ملخص البحث

تتبع الشمس وتتبع أقصى قيمة للقدرة في مصفوفات الخلايا الشمسية باستخدام متحكمات ضبابية بتقنية ال FPGA

إعداد

محمد صقر المغني

تعتبر الطاقة الشمسية احد مصادر الطاقة النظيفة والمتجددة، والأقل تلويثاً للبيئة وهي مصدر لا ينضب أكثر من كل مصادر الطاقة الأخرى المعروفة. وهذا أدى إلى ازدياد عدد التطبيقات التي تستخدم ألواح الخلايا الشمسية. وهذا يتطلب تحسين المواد والطرق المستعملة لتسخير مصدر الطاقة هذا. هناك طريقتان رئيسيتان لزيادة القدرة المتحصل عليها من الأنظمة الشمسية وهي تتبع مكان الشمس وتتبع النقطة التي تكون عندها القدرة أقصى ما يمكن. وكلا الطريقتين يحتاج إلى متحكم. هذا المتحكم إما يكون من نوع تقليدي أو من نوع ذكي مثل المتحكمات الضبابية. حيث أن للمتحكم الضبابي مميزات مثل سهولة نسبية في التصميم فهو لا يحتاج معرفة دقيقة عن المعادلات الرياضية التي تصف النظام المراد التحكم به، وهو يعمل بشكل أفضل مع الأنظمة اللاخطية. ولكي نبني هذا المتحكم فإنه يفضل استخدام تقنية مصفوفة البوابات القابلة للبرمجة الحقلية (FPGA) . وهذه الطريقة لها عدة مميزات عن المعالجات الدقيقة .

في هذا البحث تم اقتراح فكرة مطورة في هذا المجال وهي بناء متحكمَين ضبابيين على بطاقة (FPGA) حديثة من طراز (Spartan-3AN) من إنتاج شركة زيلنكس عام 2009 م . أحد المتحكمَين لجعل لوح الخلايا الشمسية يتتبع الشمس، والمتحكم الأخر ليتتبع النقطة التي تكون عندها القدرة أقصى ما يمكن. وذلك لزيادة كفاءة الخلايا الشمسية في توليد الطاقة.

تم تصميم وتطبيق متتبع الشمس والذي يتكون من : لوح من الخلايا الشمسية، مجسات، موتور الخطوة، موصلات للمداخل والمخارج، و متحكم ضبابي تم عمله على بطاقة (FPGA) وذلك لإبقاء لوح الخلايا الشمسية تواجه الشمس أغلب النهار. وتم اختبار المتحكم باستخدام برنامج السيميولينك التابع لبرنامج الماتلاب وكانت النتيجة أن استجابة المتحكم بشكل حسن.

تم تصميم وتطبيق نظام تتبع أقصى نقطة للقدرة، والمكون من لوح خلايا شمسية، محول من نوع (Buck) و متحكم ضبابي تم عمله على بطاقة (FPGA) وذلك للتحكم في فتح و إغلاق مفتاح ال (MOSFET) الموجود داخل المحول. وتم اختبار المتحكم باستخدام برنامج السيميولينك ، عن طريق مودل تم تصميمه ليحاكي سلوك أي مصفوفة خلايا شمسية ذو واجهة مرسومة للمستخدمين باستخدام برنامج (GUI) التابع لبرنامج الماتلاب وذلك لإدخال قيم المتغيرات للمصفوفة و التي يضعها المنتِج في دليل مواصفات المنتَج، لإختبار سلوك هذه المصفوفة وتأثير المتحكم عليه. وكانت نتيجة فحص المتحكم الضبابي أفضل من النتيجة المتحصل عليها من متحكم أخر تقليدي تم تطبيقه على نفس النظام.

Table of Contents

List of Tables

List of Figures :

List of Abbreviations

FLC	Fuzzy Logic Controller
FPGA	Field Programmable Gate Array
GUI	Graphical user interface
PV	Photo volatic
MPP	Maximum Power Point
VHDL	Very High Speed Integrated Circuit hardware description language
MPPT	Maximum Power Point Tracking
PLC	Programmable Logic Control
BOS	Balance of system
DC	Direct current
AC	Alternating current
PC	Personal Computer
PWM	Pulse Width Modulation
GMT	Generalized modus tollens
GMP	Generalized modus pones
TSK	Takagi-Sugeno_Kang
RCOM	Random choice of maxima
FOM	First of maxima
LOM	Last of maxima
MOM	Middle of maxima
COG	Center of gravity
MOM	Mean of maxima
BADD	Basic defuzzific ation distributions
GLSD	Generalized level set defuzzific ation
ICOG	Indexed center of gravity
SLIDE	Semi-linear defuzzification
FM	Fuzzy mean

WFM	Weighted fuzzy mean
QM	Quality method
EQM	Extended quality method
COA	Center of area
ECOA	Extended center of area
CDD	Constraint decision defuzzi_cation
FCD	Fuzzy clustering defuzzi_cation
PLC	Programmable logic controller
ASIC	Application-specific integrated circuit
IC	Integrated circuit
HDL	Hardware description language
PLD	Programmable logic devices
PLA	Programmable logic arrays
CPLD	Complex programmable logic devices
PAL	Programmable Array Logic
SPLD	Simple programmable logic devices
ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
LCD	Liquid Crystal Display
PCB	Printed Circuit Board
P&O	Perturbation and Observation
LDR	Light dependent resistor
UCF	User constraints file
DOF	Degrees of freedom

CHAPTER 1

Introduction

1.1 Introduction

Renewable energy sources play an important role in electric power generation. There are various renewable sources which used for electric power generation, such as solar energy, wind energy, geothermal etc. Solar Energy is a good choice for electric power generation, since the solar energy is directly converted into electrical energy by solar photovoltaic modules. These modules are made up of silicon cells. When many such cells are connected in series we get a solar PV module. The current rating of the modules increases when the area of the individual cells is increased, and vice versa. When many such PV modules are connected in series and parallel combinations we get a solar PV arrays, that suitable for obtaining higher power output.

The applications for solar energy are increased, and that need to improve the materials and methods used to harness this power source. Main factors that affect the efficiency of the collection process are solar cell efficiency, intensity of source radiation and storage techniques. The efficiency of a solar cell is limited by materials used in solar cell manufacturing. It is particularly difficult to make considerable improvements in the performance of the cell, and hence restricts the efficiency of the overall collection process. Therefore, the increase of the intensity of radiation received from the sun is the most attainable method of improving the performance of solar power. There are three major approaches for maximizing power extraction in solar systems. They are sun tracking, maximum power point (MPP) tracking or both. These methods needs controllers, may be intelligent such as fuzzy logic controller or conventional controller such as PID controller.

The advantage of the fuzzy logic control is that it does not strictly need any mathematical model of the plant. It is based on plant operator experience, and it is very easy to apply. Hence, many complex systems can be controlled without knowing the exact mathematical model of the plant. In addition, fuzzy logic simplifies dealing with nonlinearities in systems. The nice thing about fuzzy logic control is that the linguistic system definition becomes the control algorithm.

The most popular method of implementing fuzzy controller is using a general-purpose microprocessor or microcontroller. Microprocessor based controllers are more economical, but often face difficulties in dealing with control systems that require high processing and input/output handling speeds. Rapid advances in digital technologies have given designers the

option of implementing a controller on a variety of Programmable Logic Device (PLD), Field Programmable Gate Array (FPGA), etc. FPGA is suitable for fast implementation controller and can be programmed to do any type of digital functions. An FPGA has the ability to operate faster than a microprocessor chip. Because of the flexibility of the FPGA, additional functionality and user interface controls can be incorporated into the FPGA minimizing the requirement for additional external components.

FPGAs are programmed using Very High Speed Integrated Circuit hardware description language (VHDL) and a download cable connected to a host computer. Once they are programmed, they can be disconnected from the computer, and it will be running as stand-alone device. The FPGAs can be programmed while they run, because they can be reprogrammed in the order of microseconds. This short time means that the system will not even sense that the chip was reprogrammed [2]. Applications of FPGAs include industrial motor drivers, real time systems, digital signal processing, aerospace and defense systems, medical imaging, computer vision, speech recognition, cryptography, computer hardware emulation and a growing range of other areas.

The hardware implementation of fuzzy logic controller (FLC) on FPGA is very important because of the increasing number of fuzzy applications requiring highly parallel and high speed fuzzy processing. A significant advantage of this FLC is that it has been coded in VHDL and programmed into a single field programmable gate array (FPGA). Because this reduces the number of electronic components used to implement the controller, it enables redundancy by having multiple copies/images of the code, and yields robustness as a controller that has multiple systems capability[3]. So the FLC may implemented on FPGA and used to moves a motor attached to the solar panel to keep it toward the sun all the day. Then we must chose the kind of the motor as appropriate with the controlled system.

Many applications related to positioning systems are being implemented with stepper motors. It has some applications in Robotics, Computer peripherals, Industrial servo quality drivers and so on. One of the main advantages of stepper motors is the strong relation between electrical pulses and rotation discrete angle steps [1].

1.2 Motivation

Nowadays, fuzzy logic controllers have an efficient performance over the traditional controller researches especially in nonlinear and complex model systems. FPGA is a new key technology used in modern control hardware implementation. Modern manufactures began to apply these technologies in their applications instead of the traditional ones, due to the low cost and widely features available in these controllers. This motivated me to implement FLC on FPGA techniques. In Gaza Strip we have a big problem in electrical power generation, since our sources don't cover all people requirements, electrical power have high cost and many daily interruptions, so we need clean renewable energy sources that do not depend on others such as solar energy since Gaza Strip is 360 km2 and an excellent solar location with 2750 kWh/m2/year irradiation. The electrical problem Gaza has motivated me to investigate Solar Energy as an application to apply my fuzzy controller.

1.3 Objectives

The main objective is building a FLC using FPGA to maximizing the power output of the solar arrays.

The specific objectives include:

- Designing Fuzzy controllers.
- Realizing controllers on FPGA.
- Building a stepper motor driver.
- Tracking the sun all the day.

1.4 Literature Review

Daniel A. Pritchard had given the design, development, and evaluation of a microcomputer-based solar tracking system in 1983[4]. Then Manny studies for solar tracking appeared using the microprocessor, Saxena and Dutta in 1990[5], A. Konar and A.K. Mandal in 1991[6], and A. Zeroual in 1997 using electro-optical sensors for sun finding [7]. The microcontroller is used as base for automatic sun tracker to control a dc motor in 1998 by F. Huang [8], and used as base for maximum power point tracking controller by Eftichios

Koutroulis in 2001[9]. Hasan A. Yousef, had given the PC-based fuzzy logic controller design and Implementation to control a sun tracking system in 1999, the tracking system was driven by two permanent magnet DC motors to provide motion of the PV panels in two axes[10]. Chee-Yee Chong, in 2000 had given the process architectures for track fusion, they presented different approaches for fusing track state estimates, and compared their performance through theoretical analysis and simulations, they used the concept of multiple targets tracking because it had shown that tracking with multiple sensors can provide better performance than using a single sensor[11]. Manny studies for novel maximum power point tracking (MPPT) controller for a photovoltaic (PV) energy conversion system was proposed by Yeong Chau Kuo in 2001[12], K. K. Tse in 2002[13], and Henry Shu-Hung Chung in 2003[14], Kimiyoshi Kohayashi in 2004[15]. Z.G. Piao, proposed a solar tracking system in 2003, using DC motors, special motors like stepper motors, servo motors, real time actuators, to operate moving parts, it was highly expensive[16]. A. A.Khalil, had presented a sun tracking system in 2003, This Tracking system easy to implement and efficient for solar energy collection[17]. Manny methods was proposed to achieve the objective of maximum power point tracking (MPPT), and the active sun tracking scheme without any light sensors, S. Armstrong et al. had proposed a quantitative measure of the effectiveness MPPT efficiency in 2005, a vector methodology was used to track the direction and path of the sun throughout the day[18]. And Rong-Jong Wai. had given grid connected photovoltaic (PV) generation system with an adaptive step-perturbation method and an active sun tracking scheme in 2006[19]. Cemil Sungur had given the electromechanical control system of a photovoltaic (PV) panel tracking the sun using Programmable Logic Controls (PLC) in 2007[20]. Many FPGA-based PV systems fuzzy MPPT control was proposed, A. Messai, A. Mellit describes the hardware implementation of a two-inputs one-output digital Fuzzy Logic Controller (FLC) on a Xilinx FPGA using VHDL language in 2009[21], Cheng, Ze; Yang, Hongzhi; Sun, Ying had proposed a simple, reliable method in 2010[22]. In my study I will construct an efficient solar system with two FLCs have a good response on the same FPGA card .

1.5 Contribution

In this thesis two fuzzy logic controllers (Sun tracker and MPPT) have been implemented using modern FPGA card (Spartan-3AN, Xilinx Company, 2009), which are used to improve the efficiency of electrical power generated from photovoltaic module. These controllers have been tested using Matlab/Simulink program.

1.6 Outline Of The Thesis

The thesis is organized into seven chapters. Chapter 2 handles some basic principles of solar energy. Chapter 3 focuses on Fuzzy logic control. Chapter 4 presents full description about FPGA and VHDL software implementation. Chapter 5 presents the design of the sun tracker controller, simulation, results, and the comparison of some results with privies studies. Chapter 6 presents the design of the maximum power point tracker controller, simulation, excremental results. The last chapter concludes the design and the implementation and proposes some future work.

CHAPTER 2

Introduction to Solar Energy

2.1 Background

One of the most important problems facing the world today is the energy problem. This problem is resulted from the increase of demand for electrical energy and raised of fossil fuel prices. Another problem in the world is the global climate change has increased. As these problems alternative technologies for producing electricity have received greater attention. The most important solution was in finding other renewable energy resources [23].

2.1.1 Renewable Energy

Renewable energy is energy which comes from natural resources such as sunlight, wind, rain, tides, and geothermal heat, which are naturally replenished [24]. In its various forms, it derives directly from the sun, or from heat generated deep within the earth [25]. Figure 2.1 show that natural gas and nuclear power are expected to grow slowly over the next 40 years, at which point natural gas will start its decline. It is also hoped that a new clean energy source of fusion energy will be demonstrated at increasing scales from 2030 to 2070 which will then become commercially competitive [26].

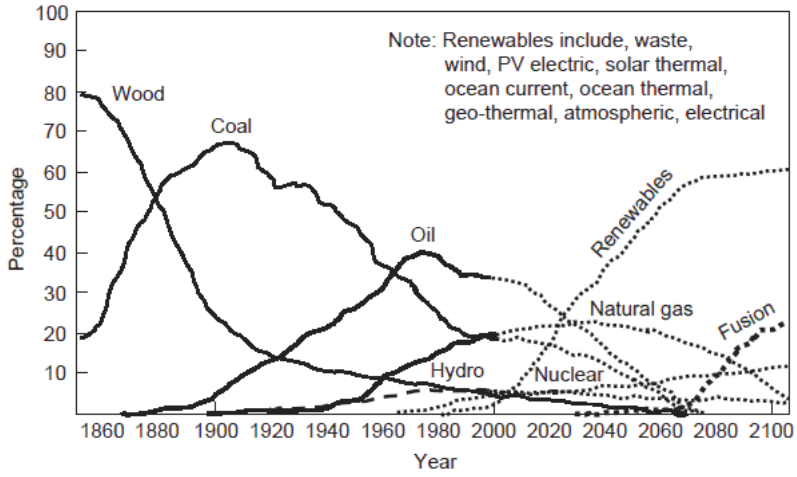


Figure 2-2: Principle of solar cells.

Renewable energy replaces conventional fuels in four distinct areas: power generation, water and space heating, transport fuels, and rural and remote areas energy services [24]. Globally, an estimated 193 million households depends on renewable energy systems [24][27].

The most important of renewable energy is solar energy; however, grid-connected PV increased the fastest of all renewable technologies, with a 60-percent annual average growth rate for the five-year period [24]. Nowadays, solar energy has been widely used in our life, and it's expected to grow up in the next years.

2.1.2 Solar Energy

It's the energy which derived from the sun through the form of solar radiation. Solar powered electrical generation relies on photovoltaic. A partial list of other solar applications includes space and water heating, solar cooking, and high temperature process heat for industrial purposes.

2.2 Photovoltaic Background:

Solar panels are made up of photovoltaic cells; it means the direct conversion of sunlight to electricity by using a semiconductor, usually made of silicon [29]. The word photovoltaic comes from the Greek meaning “light” (photo) and “electrical” (voltaic), The common abbreviation for photovoltaic is PV [30]. Bell Laboratories produced the first solar cell in 1954. The efficiency of this cell was about 5 percent. The first cells were designed for space applications, so the cost was not a major issue. Then solar cell efficiency increased continuously in the following years, and costs have decreased significantly in recent decades. The main material used in the construction of solar cells is still silicon, but other materials have been developed, either for their potential for cost reduction or their potential for high efficiency [30]. Over the last 20 years the world-wide demand for solar electric power systems has grown steadily [28]. The need for low cost electric power in isolated areas is the primary force driving the world-wide photovoltaic (PV) industry today. PV technology is simply the least-cost option for a large number of applications, such as stand-alone power systems for cottages and remote residences, remote telecommunication sites for utilities and the military, water pumping for farmers, and emergency call boxes for highways and college campuses [28].

Solar cells are converting light energy, to another form of energy, electricity. When light energy is reduced or stopped, as when the sun goes down in the evening or when a cloud

passes in front of the sun, then the conversion process stops or slows down. When the sunlight returns, the conversion process immediately resumes, this conversion without any moving parts, noise, pollution, radiation or constant maintenance. These advantages are due to the special properties of semiconductor materials that make this conversion possible. Solar cells do not store electricity; they just convert light to electricity when sunlight is available. To have electric power at night, a solar electric system needs some form of energy storage, usually batteries, to draw upon [31].

2.3 Principle of Solar Cells

Photovoltaic systems employ semiconductor cells (wafers), generally several square centimeters in size [32]. Semiconductors have four electrons in the outer shell, or orbit, on average. These electrons are called valence electrons [30]. When the sunlight hits the photovoltaic cells, part of the energy is absorbed into the semiconductor. When that happens the energy loosens the electrons which allow them to flow freely. The flow of these electrons are a current and when you put metal on the top and bottom of the photovoltaic cells, we can draw that current to use it externally, as shown in Figure 2.2 .

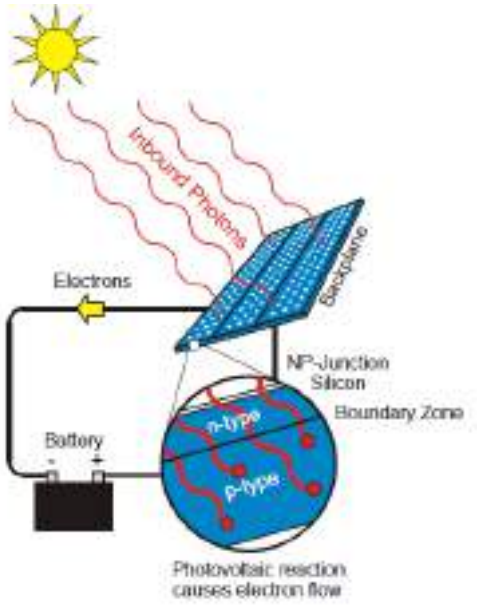


Figure 2-2: Principle of solar cells.

Numerous cells are assembled in a module to generate required power [32]. When many such cells are connected in series we get a solar PV module, the current rating of the modules depends on the area of the individual cells. For obtaining higher power output the solar PV modules are connected in series and parallel combinations forming solar PV arrays.

2.4 Types of Solar Panels

There are different types of solar panels which differ in them material, price, and efficiency, since the efficiency is the percentage of solar energy that is captured and converted into electricity. The efficiency values which gives are an average percentage of efficiency, because it's difficult to give an exact number for the different types of solar panels output [29].

- Monocrystalline Solar Panels: have efficiency approximately 18%. They are made from a large crystal of silicon, see Figure 2.3. These types of solar panels are the most efficient as in absorbing sunlight and converting it into electricity; however they are the most expensive. They do somewhat better in lower light conditions than the other types of solar panels.

Figure 2-3 Monocrystalline Solar Panels

- Polycrystalline Solar Panels: have efficiency approximately 15%. Instead of one large crystal, this type of solar panel consists of multiple amounts of smaller silicon crystals, see Figure 2.4.

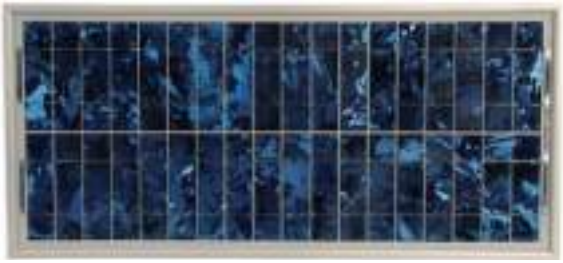


Figure 2-4 :Polycrystalline Solar Panels.

They are the most common type of solar panels on the market today. They look a lot like shattered glass. They are slightly less efficient than the monocrystalline solar panels and less expensive to produce.

- Amorphous Solar Panels: have efficiency approximately 10%. Consisting of a thin-like film made from molten silicon that is spread directly across large plates of stainless steel or similar material, see Figure 2.5. One advantage of amorphous solar panels over the other two is that they are shadow protected. That means when a part of the solar panel cells are in a shadow the solar panel continues to charge. These types of solar panels have lower efficiency than the other two types of solar panels, and the cheapest to produce. These work great on boats and other types of transportation.

Figure 2-5 :Amorphous Solar Panels.

In this thesis a polycrystalline solar panel is used.

2.5 Equivalent Circuit of a Solar Cell

From the solid-state physics point of view, the cell is basically a large area p-n diode with the junction positioned close to the top surface [32]. So an ideal solar cell may be modeled by a current source in parallel with a diode, that mathematically describes the I-V characteristic by[33]:

$$I = I_{pv,cell} - I_d = I_{pv,cell} - I_{0,cell}\left[\exp\left(\frac{qV}{akT}\right) - 1\right] \tag{2.1}$$

Where $I_{pv,cell}$ is the current generated by the incident light, I_d is the Shockley diode equation, $I_{0,cell}$ is the reverse saturation or leakage current of the diode , q is the electron charge [1.60217646 * 10^{-19}C], k is the Boltzmann constant [1.3806503 * 10^{-23}J/K], T [K] is the temperature of the p-n junction, and a is the diode ideality constant. A shunt resistance and a series resistance component are added to the model since no solar cell is ideal in practice. Figure 2.6 shows the equivalent circuit [24].

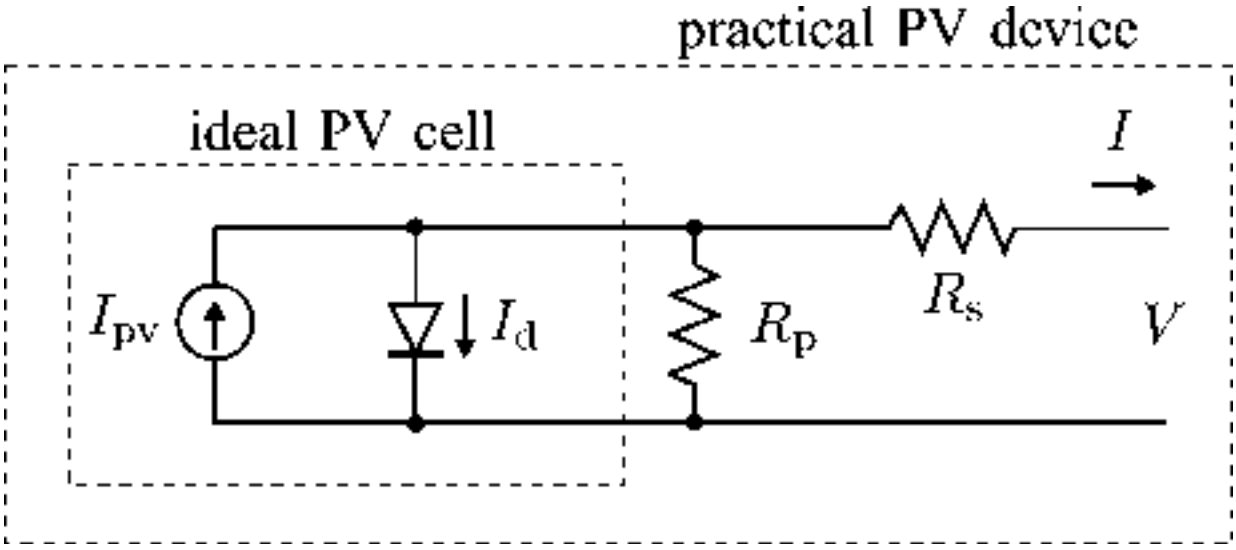


Figure 2-6:Equivalent Circuit of Solar Cell

An additional parameters is added to the basic equation to represent a practical arrays are composed of several connected photovoltaic cells and equation 2.1 becomes as [33] :

$$I = I_{pv} - I_0\left[\exp\left(\frac{V + R_s I}{V_t a}\right) - 1\right] - \frac{V + R_s I}{R_p} \tag{2.2}$$

Where I_{pv} and I_0 are the photovoltaic and saturation currents of the array and $V_t = N_s kT/q$ is the thermal voltage of the array with Ns cells connected in series R_s and R_p is the equivalent series and parallel resistance. Figure 2.7 shows the I-V curve from equation 2.2 [33].

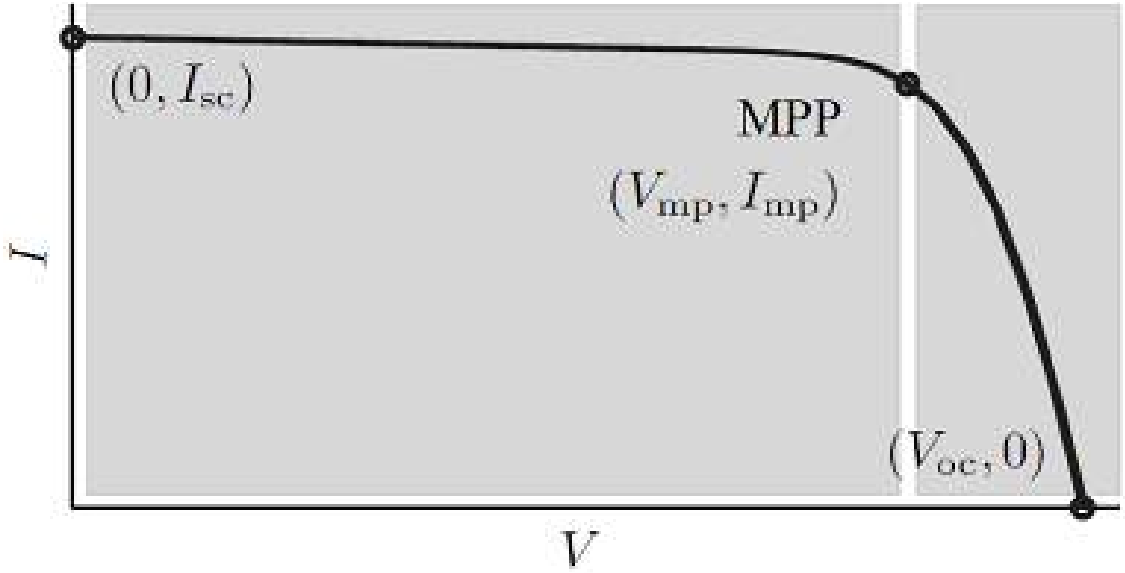


Figure 2-7 :Characteristic I-V curve of a practical photovoltaic device .

The light generated current of the photovoltaic cell I_{pv} and saturation current I_0 depend on the temperature according to the following equations[33] :

$$I_{pv} = \left(I_{pv,n} + K_I \Delta_T\right) \frac{G}{G_n} \tag{2.3}$$

$$I_0 = \frac{I_{sc,n} + K_I \Delta_T}{\exp\left(\frac{V_{oc,n} + K_V \Delta_T}{aV_t}\right) - 1} \tag{2.4}$$

Where $I_{pv,n}$ is the light-generated current at the nominal condition (usually 25 ◦C and 1000W/m^2), $\Delta T = T - T_n$ (being T and T_n the actual and nominal temperatures [K]), G [W/m^2] is the irradiation on the device surface, and G_n is the nominal irradiation.

2.6 Balance Of System (BOS):

PV modules are integrated into systems designed for specific applications. BOS is the components which added to the module. These components can be classified into four categories [28]:

- Batteries - store electricity to provide energy on demand at night or on overcast days;
- Inverters - convert the direct current (DC) output of the array or the battery into alternating current (AC).
- Rectifiers (battery chargers) - convert the AC current produced by a generator into the DC current needed to charge the batteries.
- Controllers - manage the energy storage to the battery and deliver power to the load [28], several electronic devices are used to control and modify the electrical power produced by the photovoltaic array. These include: Battery charge controllers - regulate the charge and discharge cycles of the battery; Maximum power point trackers (MPPT) as it will be explained in section 2.8; and
- Structure - required to mount or install the PV modules and other components [28]. Many of these plants are integrated with agriculture and some use innovative tracking systems that follow the sun's daily path across the sky to generate more electricity than conventional fixed-mounted systems as explained in the next section.

2.7 Sun Tracker

The sun rises each day from the east, and moves across the sky to the west. When the sun is shining, it is sending energy to us,and we can feel its heat; however and its position varies with the time of day and the seasons. Thus, if we could get a solar cell to turn and look at the sun all day, then it would be receiving the maximum amount of sunlight possible and converting it into electricity. A solar tracker is a device that is used to align a single photovoltaic panel or an array of P.V modules with the sun, so a solar tracker can improve a systems power output by keeping the sun in focus throughout the day; thus improving effectiveness of such equipment over any fixed position.

A well designed system which utilizes a tracker will reduce an initial implementation costs, since it needs fewer expensive panels due to increased efficiency. There are two general forms of tracking techniques used: dynamic tracking and fixed control algorithms. The main difference between them is the manner in which the path of the sun is determined. In the dynamic tracking system, actively searches for the sun's position at any time of day, light sensors are positioned on the tracker at various locations or in specially shaped holders. If the sun is not facing the tracker directly there will be a difference in light intensity on one light sensor compared to another and this difference can be used to determine which direction the tracker has to tilt in order to be facing the sun. On the other hand in the fixed control algorithm systems, the control system use no sensing. It does not actively find the sun's position but instead determines the position of the sun through prerecorded data for a particular site. If given the current time, day, month, and year, then the system calculates the position of the sun, so it is called 'open loop trackers'[32]. Common to both forms of tracking is the method of direction control system. In this research I will study the dynamic tracking system.

2.8 Maximum Power Point Tracking

A typical characteristic curve of PV model's current and voltage curve is shown in Figures 2.8 ,and the power and voltage curve is shown in Figures 2.9. The characteristics of a PV system vary with temperature as shown in Figures 2.10 and with irradiation as shown in Figures 2.10; there exists a single maxima power corresponding to a particular voltage and current [34].

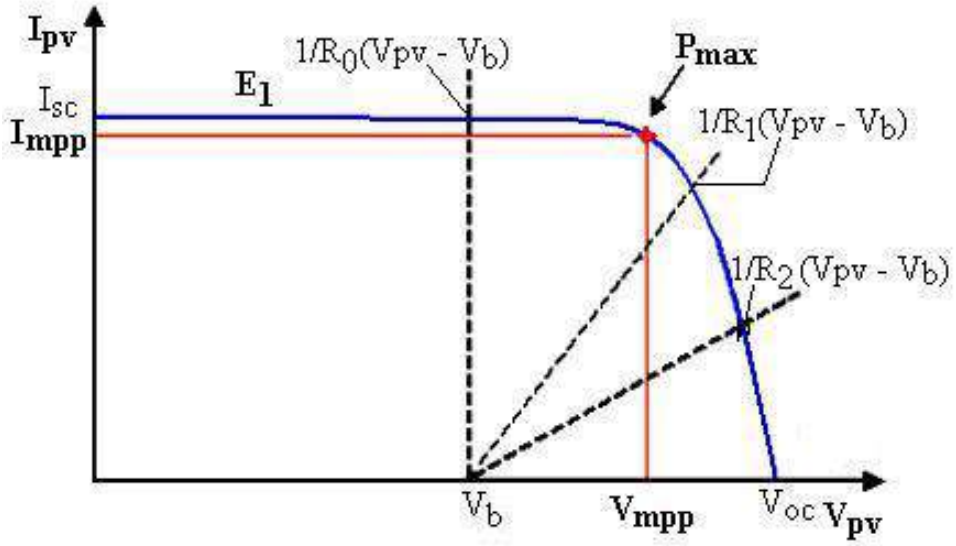


Figure 2-8 : Current-voltage characteristic of a PV module.

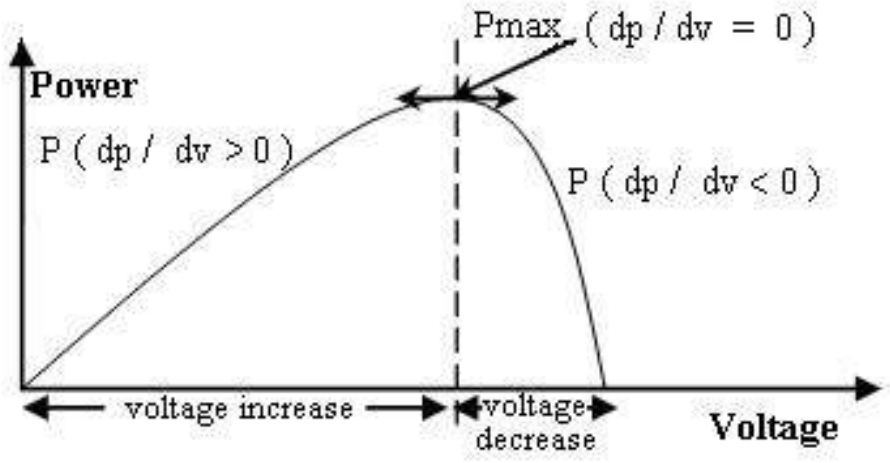


Figure 2-9: Power-voltage characteristic of a PV module

So, when a direct connection is carried out between the source and the load, the output of the PV module is seldom maximum and the operating point is not optimal. To avoid this problem, it is necessary to add an adaptation device, MPPT controller with a DC-DC converter, between the source and the load (Figure 2.12).

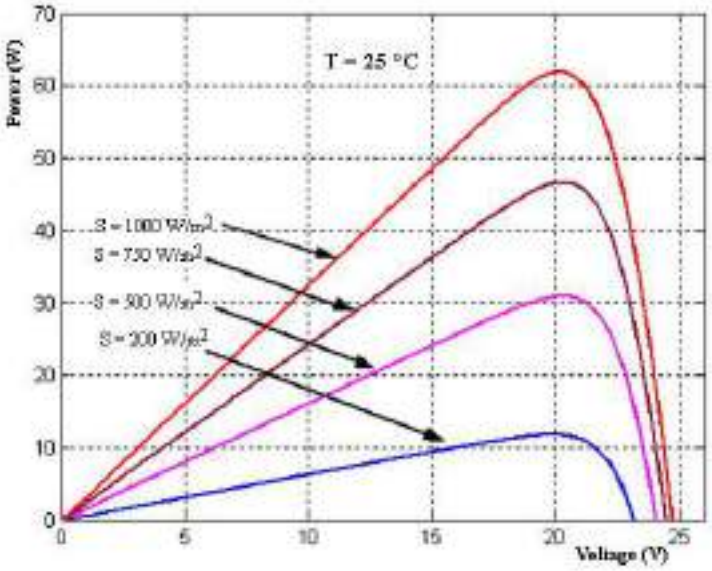


Figure 2-10: Influence of the solar radiation for constant temperature.

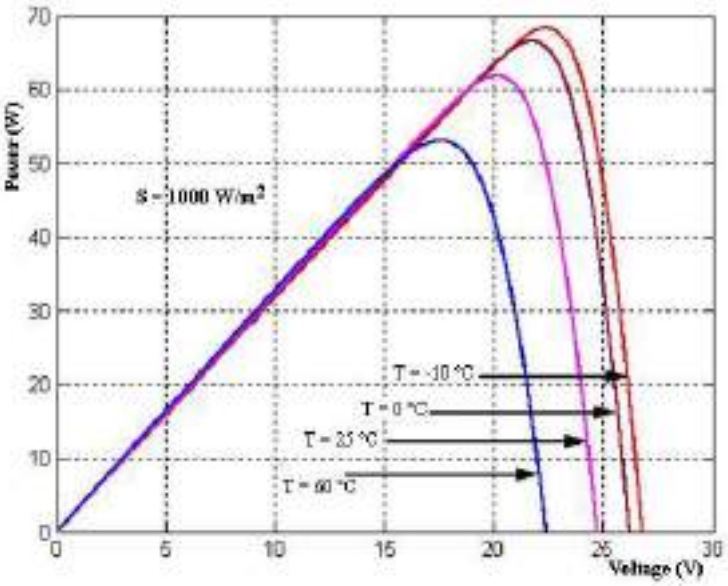


Figure 2-11:Influence of the temperature of junction for constant irradiation

Maximum power point tracker (MPPT) tracks the new modified maximum power point in its corresponding curve whenever temperature and/or insolation variation occurs. MPPT is used for extracting the maximum power from the solar PV module and transferring that power to the load. A dc/dc (step up/step down) converter acts as an interface between the load and the module.

The MPPT changing the duty cycle to keep the transfer power from the solar PV module to the load at maximum point[34].

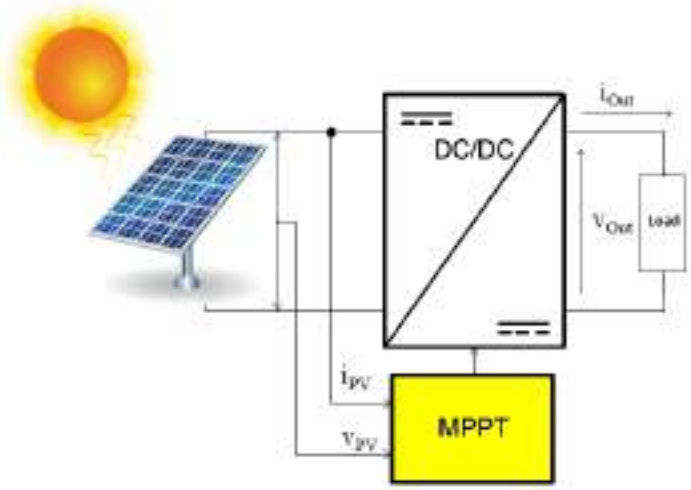


Figure 2-12 : Photovoltaic with MPPT system

2.9 Pulse Width Modulation (PWM)

PWM signals are pulse trains with fixed frequency and magnitude and variable pulse width. However, the width of the pulses (duty cycle) changes from pulse to pulse according to a modulating signal as illustrated in Figure 2.13. When a PWM signal is applied to the gate of a power transistor, it causes the turn on and turn off intervals of the transistor to change from one PWM period to another according to the same modulating signal[65].

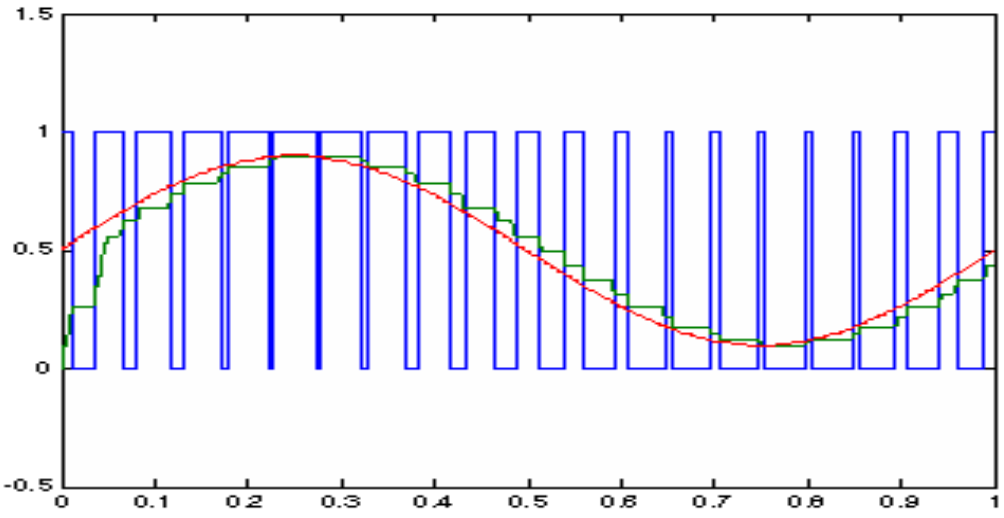


Figure 2-13: **Pulse width modulation waveforms.**

2.10 Buck Converter

A buck converter is a step-down DC to DC converter. The operation of the buck converter is fairly simple, with an inductor and two switches (transistor and diode) that control the current of the inductor as shown in Figure 2.14 .

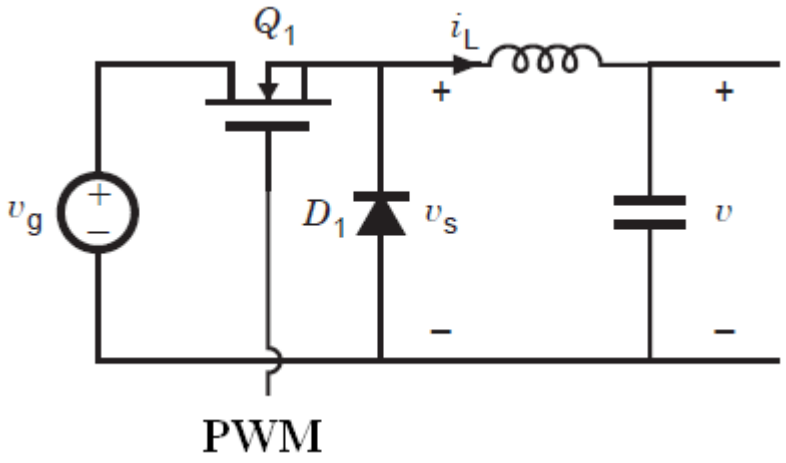


Figure 2-14: Buck converter

It alternates between connecting the inductor to source voltage to store energy in the inductor when the PWM signal is high and discharging the inductor into the load when the PWM signal is low. When the duty cycle is in ON state, The diode become as reversed biased and the inductor will deliver current and switch conducts inductor current. The current through the inductor increase, as the source voltage would be greater. The energy stored in inductor increased when the current increase, and the inductor acquires energy. Capacitor will provides smooth out of inductor current changes into a stable voltage at output voltage. When the duty cycle is in OFF state, The diode is ON and the inductor will maintains current to load. Because of inductive energy storage, inductor current will continues to flow. While inductor releases current storage, it will flow to the load and provides voltage to the circuit. The diode is forward biased. The current flow through the diode which is inductor voltage is equal with negative output voltage.

CHAPTER 3

Fuzzy Logic Control

3.1 Introduction

In 1965, Professor L.A. Zadeh of University California, Berkley, presented his first paper[35] on fuzzy set theory[36]. He developed a mathematical way of looking at vagueness. The objective has been to make computers think like humans and to enable computing with words [37],[38]. However, during its early years, it was met with a lot of criticisms, some of which are from Prof. Zadeh's colleagues themselves. In 1974, Mamdani published the first paper for fuzzy applications [39]. In 1985, Takagi and Sugeno published the paper of fuzzy systems [40]. In about 1970, fuzzy theory began to produce results in Japan, Europe, and China. Fuzzy control has found applications in cement kinks, papermaking machines, and polymerization reactors[35]. The number of applications of Fuzzy Logic Controllers (FLCs) has dramatically increased. Initially outlined by Zadeh and explored by Mamdani, FLC applications exhibited their first industrial and commercial growth in Japan almost a decade later. Since then, many Japanese companies have offered consumer-oriented products enhanced by FLC technology, such as Canon's camera autofocus control, Honda's and Nissan automatic transmission, Mitsubishi's room air conditioner control, Panasonic's clothes washer control, and Toshiba's elevator control[41]. In October of 1993, at the Tokyo Motor show, Mitsubishi had a computer which imitates the information processing in a driver's brain. The computer studies the driver's normal driving habits and selects a response from different situations that could happen. If a built in radar system detects an object in the road, then the fuzzy logic system would decide whether or not the driver was aware of the obstacle based on previous driving patterns. If the driver does not respond as the computer predicted, then the computer can automatically take control of the brakes to avoid a collision.[42] In 1994 Japan exported products using fuzzy logic totaling 35 billion dollar[43]. Nowadays, fuzzy controllers are also used to control consumer products, such as dishwashers, washing machines, video cameras, and rice cookers [35].

3.2 Fuzzy Sets

3.2.1 Basic Concepts

In crisp sets, an element in the universe has a well defined membership or non membership to a given set. Membership to a crisp set E can be defined through a membership function defined for every element x of the universe as

$$\mu_E(x) = \begin{cases} 1 & x \in E \\ 0 & x \notin E \end{cases}$$

But for an element in a universe with fuzzy sets, the membership function can take any value between 0 and 1. This transition among various degrees of membership can be thought of as conforming to the fact that the boundaries of the fuzzy sets are vague and ambiguous. An example of a graphic for the membership function of a crisp set is illustrated in Figure 3.1.

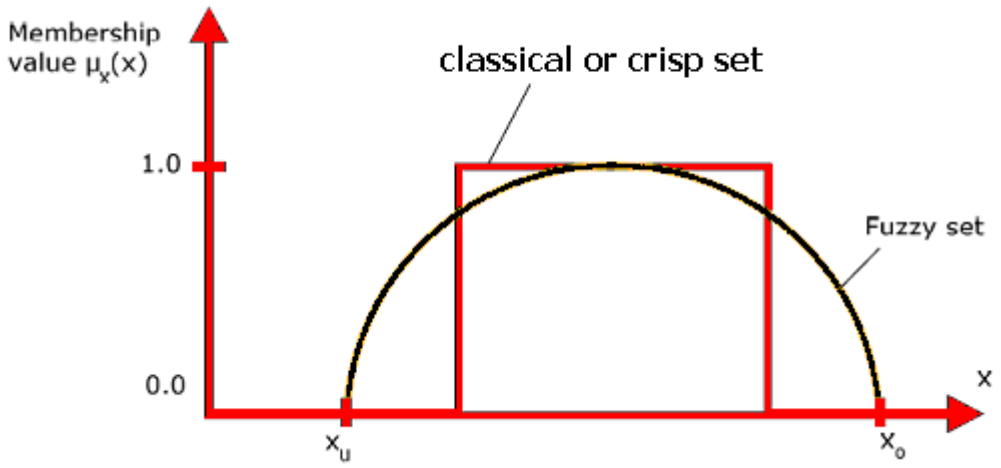


Figure 3-1 : Fuzzy and Classical sets

Fuzzy membership of an element from the universe in this set is measured by a function that attempts to describe vagueness. In fuzzy logic, linguistic variables take on linguistic values which are words with associated degrees of membership in the set. Thus, instead of a variable temperature assuming a numerical value of 70 C^o, it is treated as a linguistic variable that may assume, for example, linguistic values of “hot” with a degree of membership of 0.92, "very cool" with a degree of 0.06, or "very hot" with a degree of 0.7. Each linguistic term is associated with a fuzzy set, each of which has a defined membership function.

Formally, a fuzzy set is defined as a set of pairs where each element in the universe F has a degree of membership associated with it:

$E = \{(x, \mu E(x)) \mid x \in F, \mu E(x) \in [0, 1]\}$

The value $\mu_E(x)$ is the degree of membership of object x to the fuzzy set E where $\mu_E(x) = 0$ means that x does not belong at all to the set, while $\mu_E(x) = 1$ means that the element is totally within the set [44].

3.2.2 Basic Operations on Fuzzy Sets

Every fuzzy set can be represented by its membership function. The shape of membership function depends on the application and can be monotonic, triangular, trapezoidal or bellshaped as shown in Figure 3.2 [50].

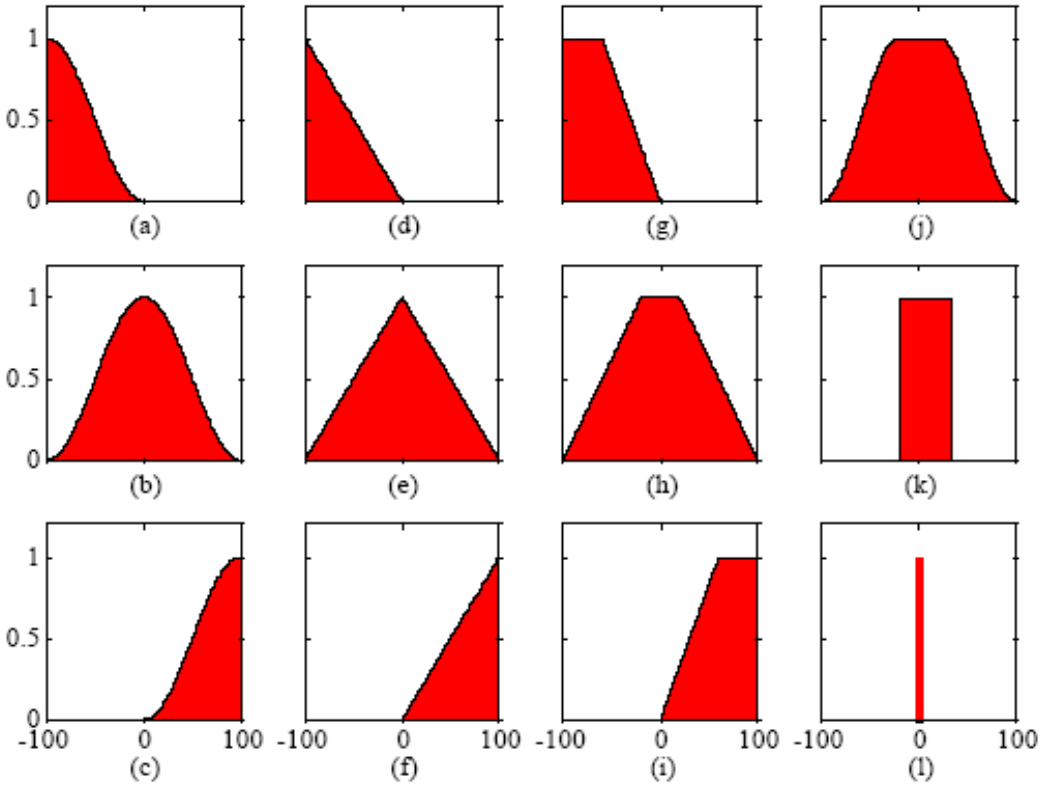


Figure 3-2 : Different shapes of membership functions (a) s_function, (b) π_function, (c) z_function, (d-f) triangular versions, (g-i) trapezoidal versions, (J) flat π_function, (k) rectangle, (L) singleton.

The membership function could be defined as a graphical representation of the quantity of participation of the inputs. It links a value with each of the inputs parameters that are treated, defines functional overlap amongst inputs, and finally defines an output parameter. The rules usually take the input membership parameters as features to establish their weight over the 'fuzzy output sets' of the final output response. Once the functions are deduct, scaled, and combined, they have to be defuzzified into a crisp output which leads the application. There are some different memberships functions linked to each input and output parameter [45].

As an example to represent the property: warm of the linguistic variable "temperature" shown in Figure 3.3. If the measured temperature in one system is x, then the level of membership of

x in the fuzzy set positive small is given by μ(x) and it is 0.7. We can say that the level of truth for the proposition:" The temperature x is warm is 0.7 or 70%".

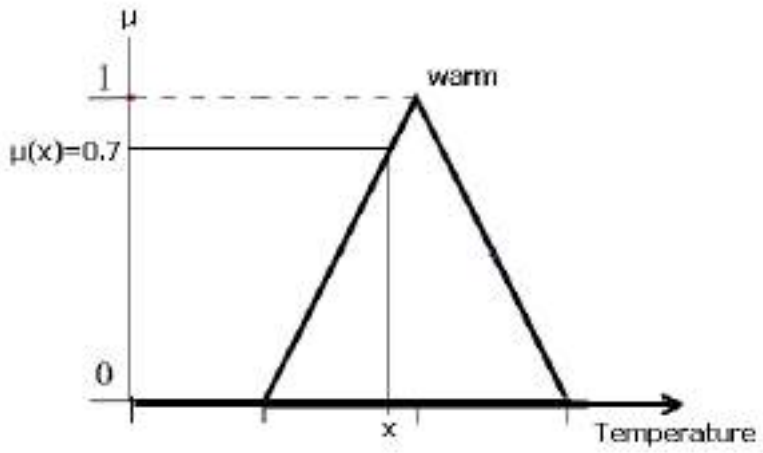


Figure 3-3: **Membership function example (warm temperature)**

One of the first steps in every fuzzy application is to define the universe of discourse (dynamic range) for every linguistic variable. The set of terms: T(temperature) can be characterized as fuzzy sets whose membership functions are shown in Figure 3.4. Every fuzzy set in a universe of discourse represents one linguistic value or label.

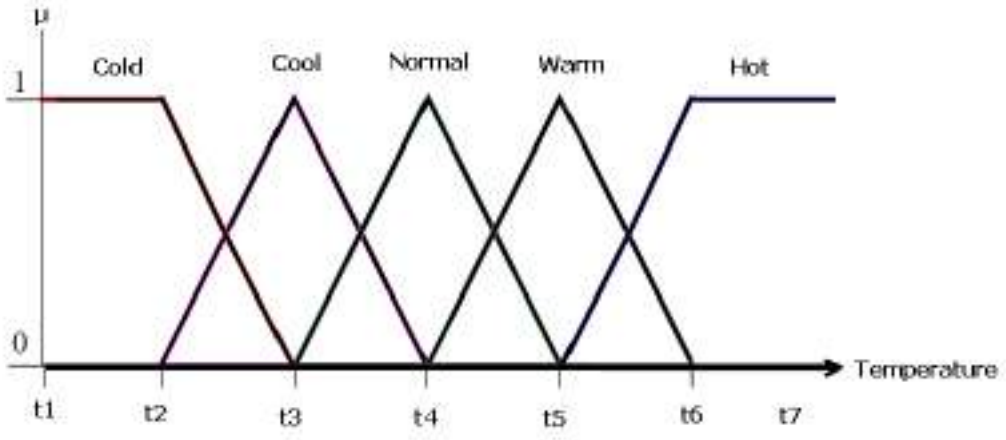


Figure 3-4: Universe of discourse for linguistic variable: temperature.

Basic operations on sets in crisp set theory are the set complement, set intersection, and set union. Fuzzy set operations are very important because they can describe intersections between variables For a given element x of the universe, the following function theoretic

operations for the set theoretic operations of complement, intersection, and union are defined [46]:

Complement (NOT):

Consider a fuzzy set A in universe X. Its complement A as displayed in Figure 3.5.

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x)$$

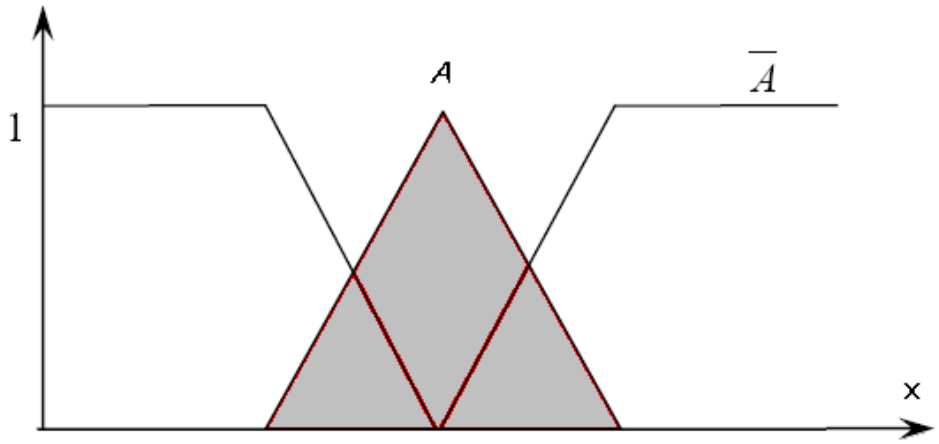


Figure 3-5 : Complement of fuzzy sets A

Intersection (AND):

Consider two fuzzy sets A and B, as shown in Figure 3.6, in the same universe X.

$$\mu_{A\cap B}(x) = \mu_A(x) \wedge \mu_B(x) = \min[\mu_A(x).\mu_B(x)] \qquad \forall x \in X$$

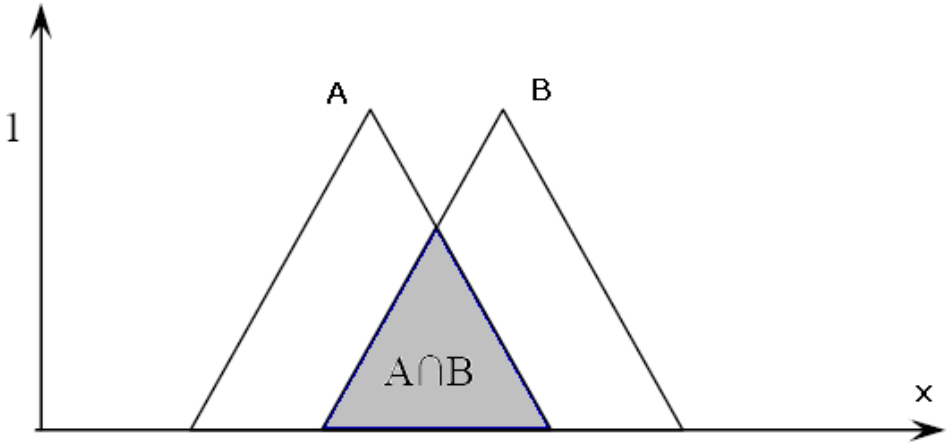


Figure 3-6 : Intersection of fuzzy sets A and B

The connective operator minimum (AND) used mostly in fuzzy logic control .

Union (OR):

Consider two fuzzy sets A and B in the same universe X. AU B is the whole area covered by the sets as shown in Figure 3.7

$$\mu_{A\cup B}(x) = \mu_A(x) \wedge \mu_B(x) = \max[\mu_A(x).\mu_B(x)] \qquad \forall x \in X$$

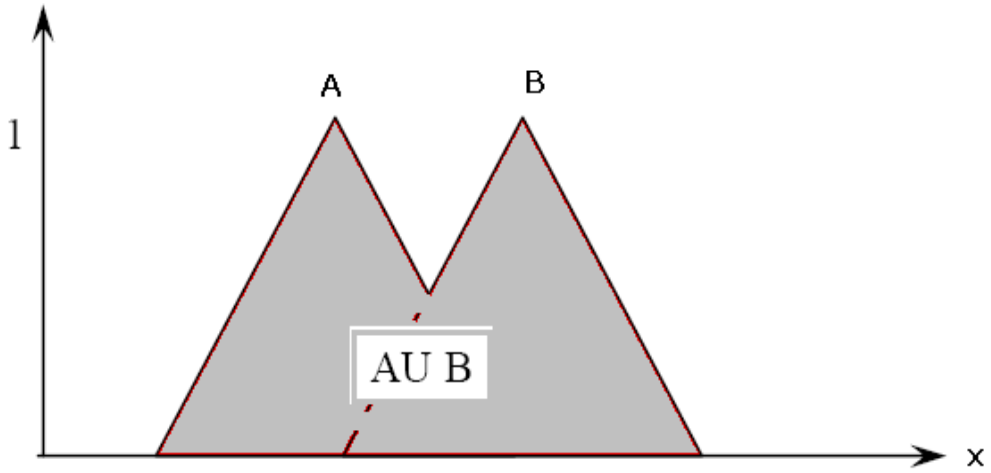


Figure 3-7: Union of fuzzy sets A and B

3.3 Fuzzy Logic :

Fuzzy Logic is a control system methodology designed for solving problems implemented in a widely range of systems such as: simple and small devices, microcontrollers or, on the other hand, large systems joined to networks, workstations or normal control systems. It may be developed in hardware, software, or both in combination. [47]

Fuzzy logic extends conventional Boolean logic to handle the concept of the partial truth the values falling between “totally true” and “totally false”. These values are dealt with using degree of membership of an element to a set. The degree of membership can take any real value in the interval [0, 1]. Fuzzy logic makes it possible to imitate the behavior of human logic, which tends to work with “fuzzy” concepts of truth. [44].

Fuzzy Logic shows a usual rule-based IF condition AND condition THEN action. It approaches to a solving control problem rather than intending to model a system based on

math's. The Fuzzy Logic model is based on empiric experience, relying on a number of samples rather than some technical understanding of the problem to be solved [47].

3.4 Fuzzy Logic Controller Structure

The basic parts of every fuzzy controller are displayed in the following Figure 3.8 [48] .The fuzzy logic controller (FLC) is composed of a fuzzification interface, knowledge base, inference engine, and defuzzification interface.

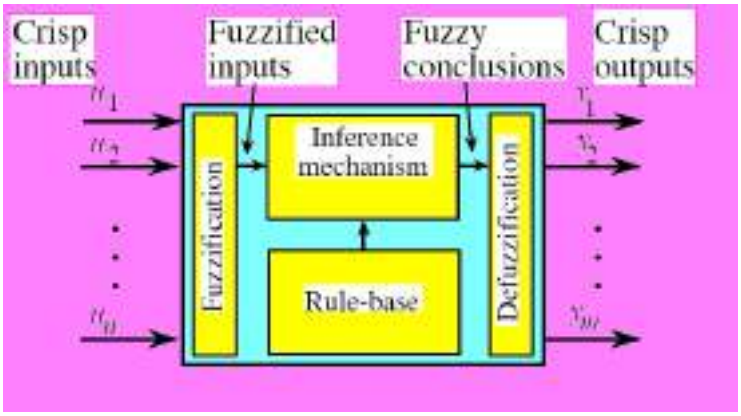


Figure 3-8 : Basic parts of a Fuzzy Controller

The fuzzifier maps the input crisp numbers into the fuzzy sets to obtain degrees of membership. It is needed in order to activate rules, which are in terms of the linguistic variables. The inference engine of the FLC maps the antecedent fuzzy (IF part) sets into consequent fuzzy sets (THEN part). This engine handles the way in which the rules are combined. The defuzzifier maps output fuzzy sets into a crisp number, which becomes the output of the FLC [49].

3.4.1 Fuzzification

The first step in fuzzy logic processing the crisp inputs are transformed into fuzzy inputs (Figure 3.9). This transformation is called fuzzification. The system must turn numeric values into language and corresponding domains to allow the fuzzy inference engine to inference to transform crisp input into fuzzy input, membership functions must first be defined for each input. Once membership functions are defined, fuzzification takes a real time input value, such as temperature, and compares it with the stored membership function information to produce

fuzzy input values. Fuzzification plays an important role in dealing with uncertain information which might be objective in nature.

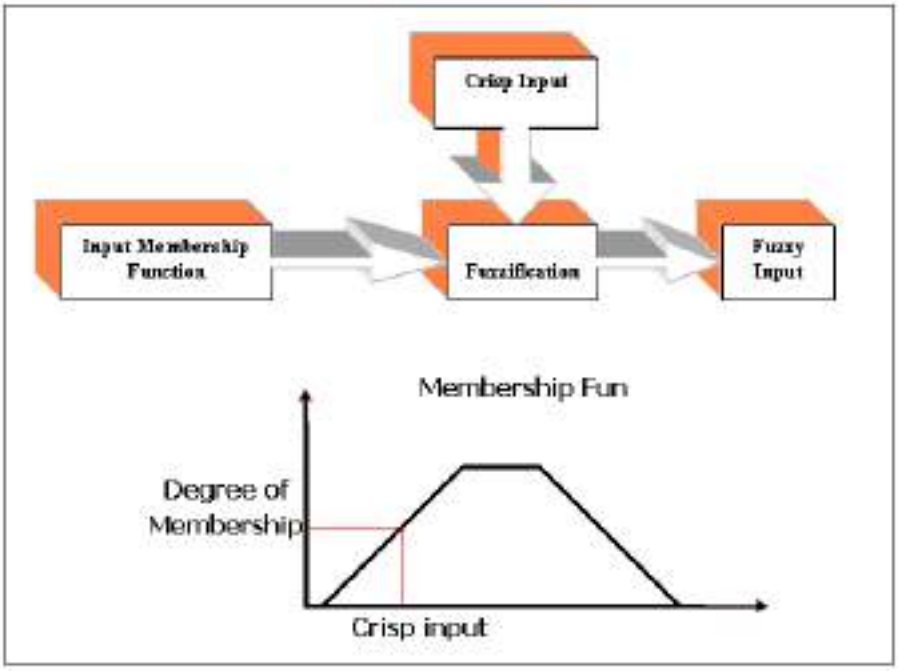


Figure 3-9 : Fuzzification

3.4.2 Knowledge Base:

Knowledge base is the inference basis for fuzzy control. It defines all relevant language control rules and parameters. The knowledge base is the core of a fuzzy control system.

The knowledge base of Fuzzy Logic Controller (FLC) is comprised of two parts [46]:

1. Database
2. Rule base

There are four principal design parameters in the data base for a fuzzy logic controller:

1. Discretization
2. Normalization of universe of discourse,
3. Fuzzy partition of input and output spaces,
4. Membership functions of primary fuzzy set.

Basically a linguistic controller contains rules in the (if-then) format [50]. The Rule base is the cornerstone of the fuzzy model. The expert knowledge, which is assumed to be given as a number of if-then rules, is stored in a fuzzy rule base [35]. The rules may use several variables

both in the condition and the conclusion of the rules. The controllers can therefore be applied to both multi-input-multi-output (MIMO)problems and single-input-single-output (SISO) problems [50].

The rules base is usually built based on the following sources:

- Expert experience and control engineering knowledge: The fuzzy control rule is based on information obtained by a controlled system. Experience rules are the most important part of fuzzy control.
- Operators' control actions: observation of human controller's actions in terms of input-output operating data.
- The fuzzy model of a process: linguistic description of the dynamic characteristics of a process.
- System self-learning: The system has an off-line learning method and builds a rule base with the help of other algorithms. self-organising controller is an example of a controller that finds the rules itself. Neural networks is another possibility.

These four modes are not mutually exclusive, and it is necessary to combine them to obtain an effective system [51].

Fuzzy control rules have two types of presentations. The first is a state evaluation type, and the second is an object evaluation type.

a) A state evaluation type that evaluates the system state at time (t) and calculates the fuzzy control action at certain point in time using the control rule and database input variables[52]. The State variables are in the antecedent part of rules and control variables are in the consequent part [51].

The following rule is presented as a simple
conditional statement:

$$R_t \text{ IF } x \text{ is } A_t \text{ THEN } y \text{ is } B_t$$

where:

- t is the statement number.

- IF is the statement antecedent proposing the conditions to determine whether the statement is true.
- THEN is the statement consequence representing the inference result according to the conditions.
- Antecedent x is the input variable of the fuzzy system and is used to measure the system state.
- Consequent y is the output variable of the fuzzy system and is used to control the system.
- A_t and B_t are fuzzy sets described by membership functions.

Another type of rules, where the consequent is a crisp function of the input variables, rather than a fuzzy proposition [54], its will explained in details in the next section.

$$R_t \text{ IF } x \text{ is } A_t \text{ THEN } y = f(x_t)$$

b) Object evaluation type (predictive fuzzy control) that predicts current and future control actions. It determines whether the control object is achieved and then decides whether to output a control command[52]. A typical rule is described as [55]:

$$R_t\text{: if } (z \text{ is } C_t \rightarrow (x \text{ is } A_t \text{ and } y \text{ is } B_t)) \text{ then } z \text{ is } C_t.$$

were x and y are performance indices for the evaluation and z is control command.
The rule is interpreted as: if the performance index x is A_t and index y is B_t when a control command z_t is C_t, then this rule is selected, and the control command C_t is taken to be the output of the controller.

3.4.3 Fuzzy Inference Engine

Fuzzy inference is the computation of fuzzy rules which represent the relationship between observations and actions. since the precondition (IF-part) can consist of multiple conditions linked together with AND or OR conjunctions. Conditions may be negated with a NOT [56]. In general, the inference is a process to obtain new information by using existing knowledge. The fuzzy inference engine performs the actual decisionmaking process[52].
The engine has two key inference methods [52]:

1. generalized modus tollens (GMT): is object-oriented inverse fuzzy theory. For example:

   Fact: y is $\bar{b}$

   Rule: If x is a then y is b

   Result: x is $\bar{a}$

2. generalized modus pones (GMP): is forwarding linking inference modus. For example:

   Fact: x is a

   Rule: If x is a, then y is b

   Result: y is b

The modus ponens is used in the forward inference and the modus tollens is in the backward one.

In GMP, when data is input, the output can be inferred according to rules; therefore, GMP is applicable for a fuzzy control inference mechanism.

There are a lot of inference methods which deals with fuzzy inference like : Mamdani method, Larsen method, Tsukamoto method, and the Takagi-Sugeno_Kang (TSK) method. The most important and widely used in fuzzy controllers are the Mamdani and Takagi-Sugeno methods[57].

3.4.3.1 Mamdani Fuzzy Model

Mamdani Fuzzy model is used widely by fuzzy system designers [58]. It was proposed in 1975, by Professor Ebrahim Mamdani of London University, as the very first attempt to control a steam engine and boiler combination by a set of fuzzy if-then rules to mimic a successful human operator who can express his/her expertise. Mamdani fuzzy logic use the linguistic variables to describe the process states and use these variables as input to control rules. Input variables are the basic of system. The terms of the linguistic variables are fuzzy sets with certain shape. It usually uses the trapezoidal or triangular fuzzy set but other shapes are possible[59].

Mamdani Fuzzy model can be formed in five steps[59]:

1. Fuzzify input: The first step is to take the crisp inputs and determine the degree to which these inputs belong to each of the appropriate fuzzy sets.

Example 3.1:

Our example is built on three rules, each depends on resolving the inputs into a number of different fuzzy linguistic sets: service is poor, service is good, food is rancid, food is delicious, and so on. The inputs must be fuzzified according to each of these linguistic sets. For example, in Figure 3.10 if we rated the food as an 8, which, given our graphical definition of delicious, corresponds to $\mu = 0.7$ for the delicious membership function[60].

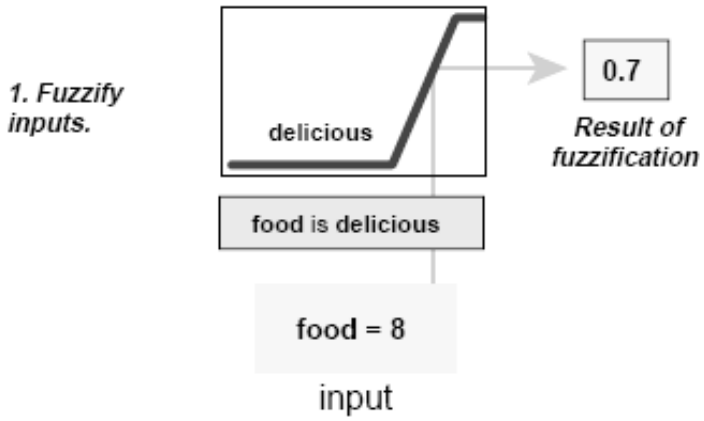


Figure 3-10 : **Fuzzify input**

2. Apply fuzzy operator to multiple part antecedents: use the fuzzy logic operators and resolve the antecedent to a single number between 0 and 1. It is the degree of support for the rule. If the fuzzy operator OR is used the fuzzy expert systems make use of the classical fuzzy operation union:

$$\mu_{A\cup B}(x) = \max [\mu_A(x), \mu_B(x)]$$

Similarly, in order to evaluate the conjunction of the rule antecedents, we apply the AND fuzzy operation intersection:

$$\mu_{A\cap B}(x) = \min [\mu_A(x), \mu_B(x)]$$

Figure 3.11 is an example of the OR operator *max* at work. The two different pieces of the antecedent (service is excellent and food is delicious) yielded the fuzzy membership values 0.0 and 0.7 respectively. The fuzzy OR operator simply selects the maximum of the two values, 0.7.

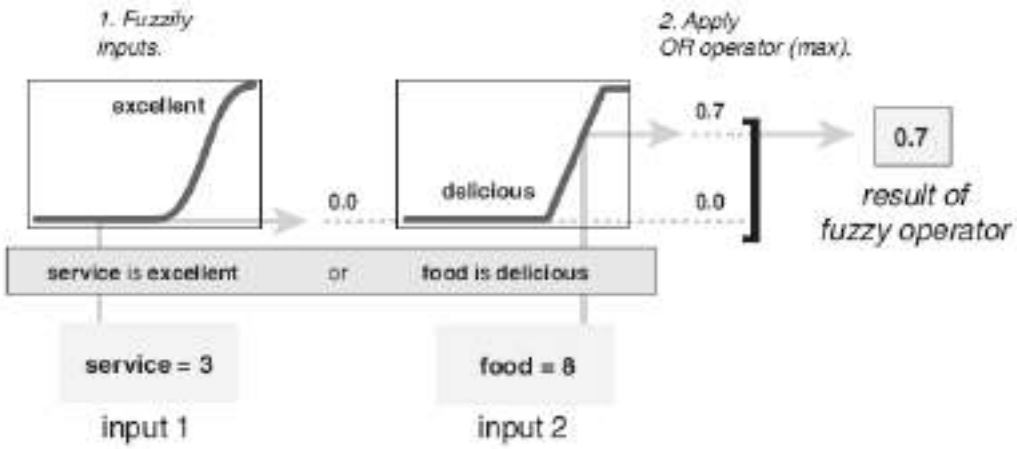


Figure 3-11: Appling OR operator to multiple part antecedents

3. Apply implication method: use the degree of support for the rule to shape the output fuzzy set. The input for the implication process is a single number given by the antecedent, and the output is a fuzzy set. Implication is implemented for each rule. Two built-in methods are supported, and they are the same functions that are used by the AND method: min (minimum), which truncates the output fuzzy set, and prod (product), which scales the output fuzzy set. see Figure 3.12.

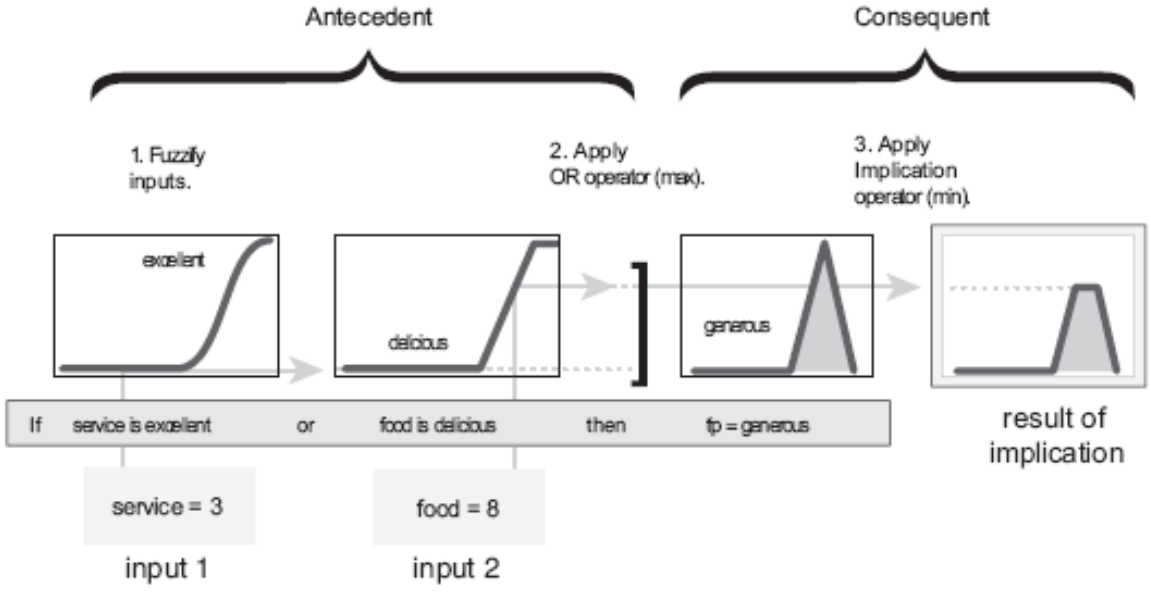


Figure 3-12: Appling implication method

4. Aggregate all Outputs: The resulting output membership functions are added together yielding a sort of probability function. This function can then be used to estimate the expected value of the output variable.

In Figure 3.13, all three rules have been placed together to show how the output of each rule is combined, or aggregated, into a single fuzzy set whose membership function assigns a weighting for every output (tip) value.

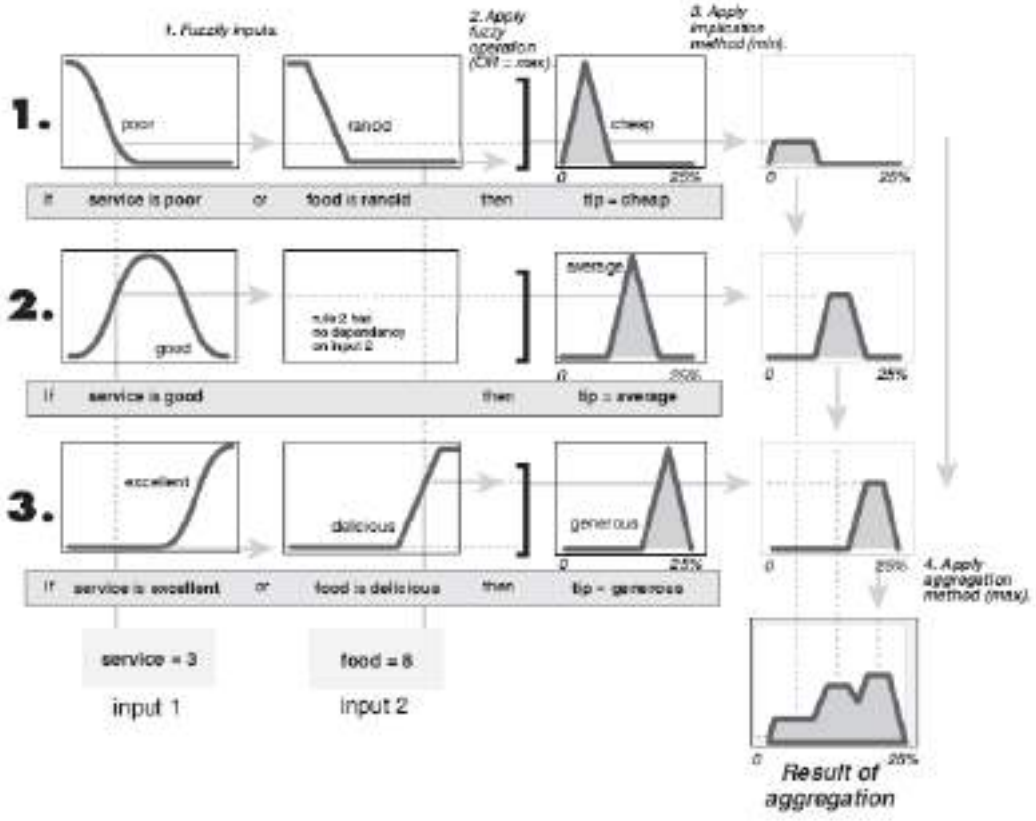


Figure 3-13 Aggregating all Outputs

5. Defuzzification.: defuzzify the aggregate output fuzzy set into a single number. See Figure 3.14. This step will be explained in details in section 3.4.4.

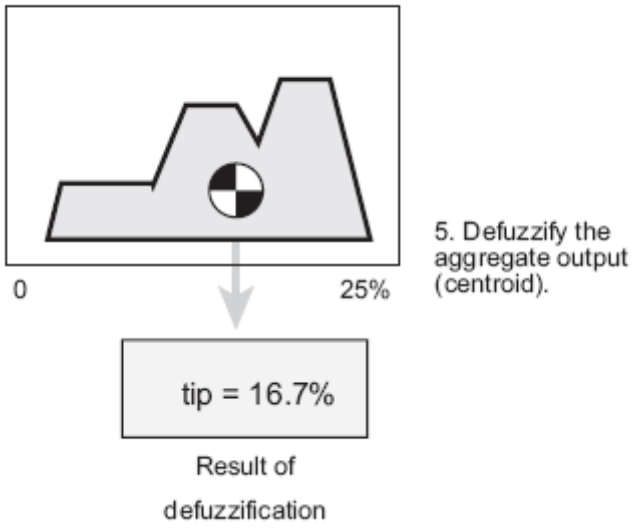


Figure 3-14: Defuzzification

Shown in Figure 3.15 is the real full-size fuzzy inference diagram for our example.

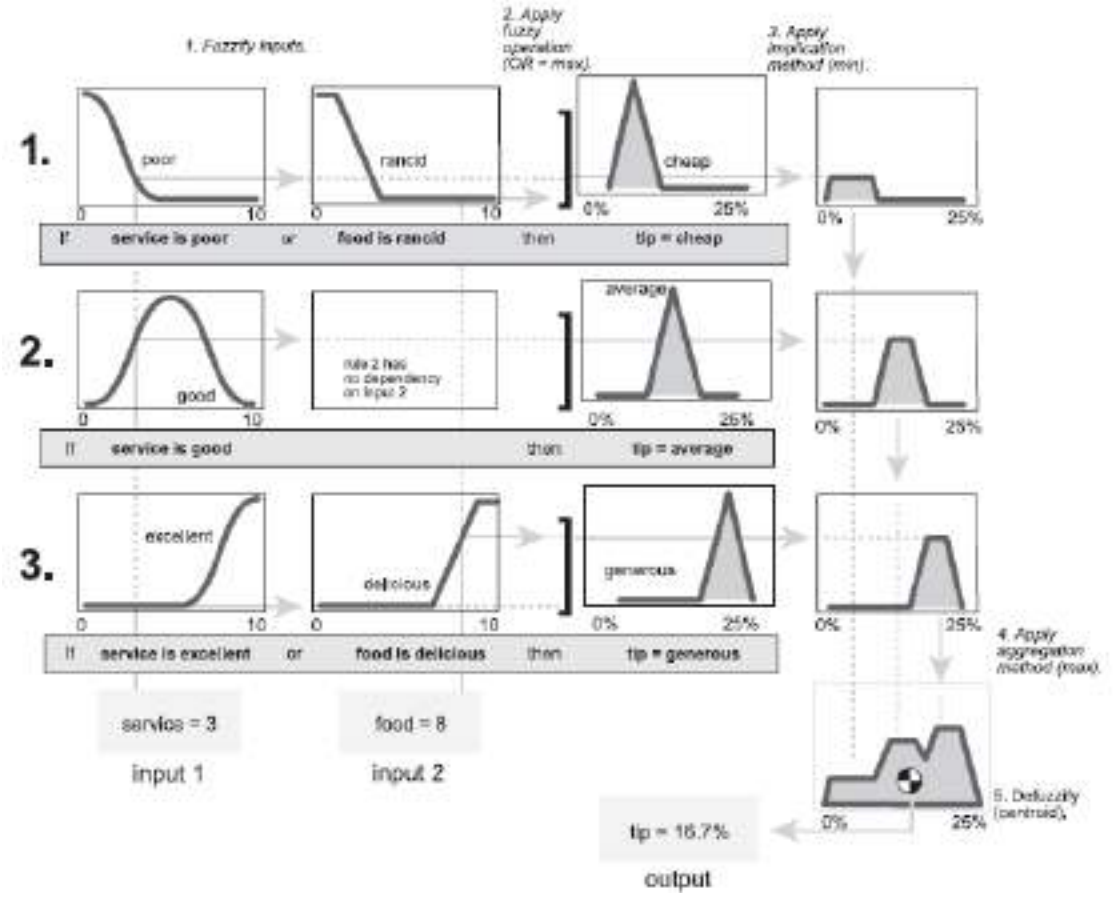


Figure 3-15 : Full-size fuzzy inference diagram

Advantages of the Mamdani Fuzzy method:

- It is intuitive and simple to build.
- It is widely used for second order systems with both linear and nonlinear characteristics.
- It has widespread acceptance.
- It is well suited to human feeling.

Disadvantages of the Mamdani Fuzzy method:

- It is only suited to the long delay system, such as the temperature control system, since it is too simple to control the process quickly.
- It needs additional device to improve the efficiency, when it controls the high frequent input system.

3.4.3.2 Sugeno Method

The Sugeno fuzzy inference was proposed by Takagi, Sugeno, and Kang (Sugeno and Kang (1988)), (Takagi and Sugeno (1985)). This model suggested to use a single spike, a singleton, as the membership function of the rule consequent, and provides a powerful tool for modeling complex nonlinear systems[59]. The mamdani model is typically used in knowledge-based (expert) systems, but the Takagi_Sugeno model used in data-driven systems[61]. The first two parts of Mamdani fuzzy inference process, fuzzifying the inputs and applying the fuzzy operator, are exactly the same. The main difference between Mamdani and Sugeno is that the Sugeno conclusions are represents by functions, Sugeno changed only a rule consequent.

A typical form of fuzzy rule in a Sugeno fuzzy model for a system has two inputs x and y and one output z is :

If x is A and y is B then z = f(x,y)

Where A and B are Fuzzy sets in the antecedent, while z= f(x,y) is a crisp function in the consequent. Usually f(x,y) is a polynomial as $z = ax + by + c$, but it can be any function as long as it can appropriately describe the output of the system within the fuzzy region specified by the antecedent of the rule. When f(x,y) is a first –order polynomial, the resulting Fuzzy inference system is called a first order Sugeno Fuzzy model, which was originally proposed in

Sugeno and Kang (1988), Takagi and Sugeno (1985). When f is a constant, or the output level *z* is a constant *(a=b=0),* we then have a zero order Sugeno fuzzy model, which can be viewed either as a special case of the Mamdani fuzzy inference system, in which each rule's consequent is specified by a fuzzy singleton .

The output level z_i of each rule is weighted by the firing strength w_i of the rule. For example, for an AND rule with Input 1 = x and Input 2 = y, the firing strength is

$$w_i = \text{And\_Method}\,(F_1(x), F_2(y))$$

where $F_{1,2}$ (.) are the membership functions for Inputs 1 and 2 [60]. The final output u is determined as a weighted mean value over all rules according to

$$u = \frac{\sum_{i=1}^{n} w_i z_i}{\sum_{i=1}^{n} w_i}$$

The effort of performing a defuzzification is saved, as the crisp value u is directly determined by the inference operation and this makes this method attractive.

A Sugeno rule operates as shown in Figure 3.16[60].

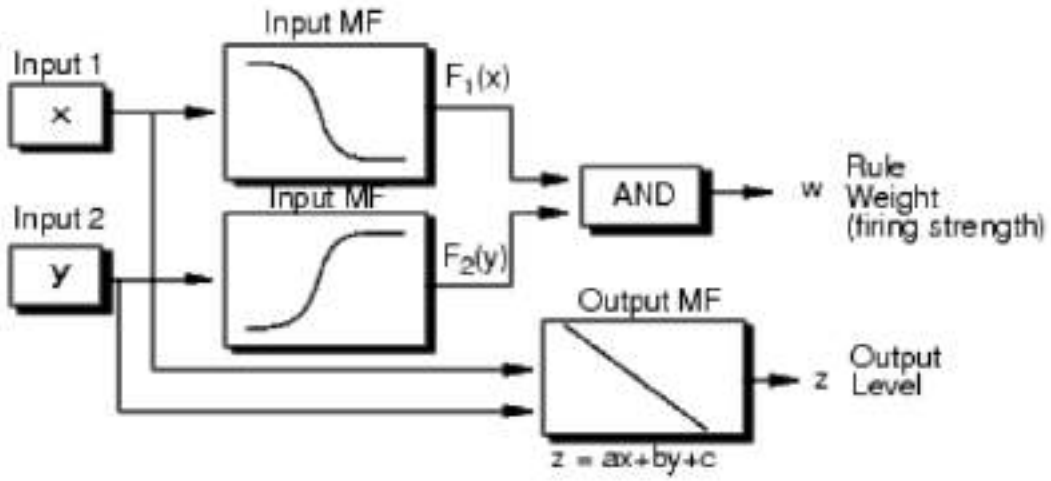


Figure 3-16 : Sugeno rule Operation

Example 3.2: Figure 3.17 shows the fuzzy tipping model developed in previous Example 3.1 adapted for use as a Sugeno system. Fortunately, it is frequently the case that singleton output functions are completely sufficient for the needs of a given problem[60].

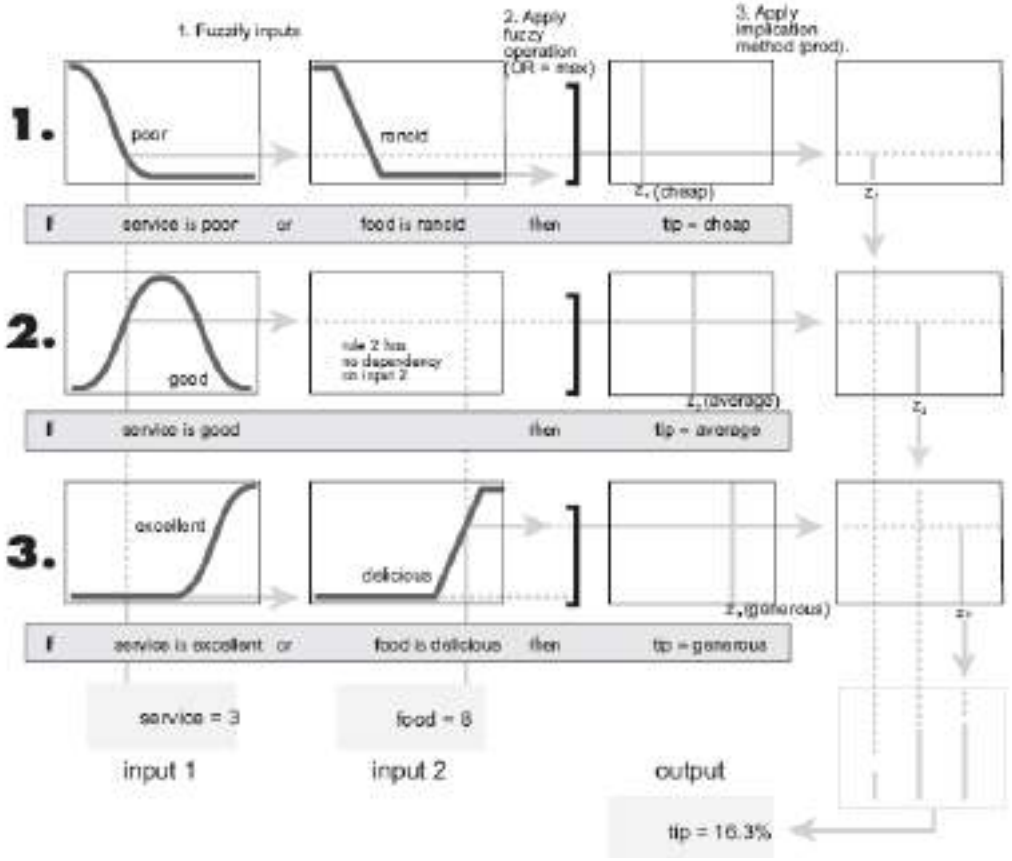


Figure 3-17: Fuzzy tipping model

The advantages of Takagi-Sugeno Model:

- It is computationally efficient.
- It works well with linear techniques (e.g., PID control).
- It works well with optimization and adaptive techniques.
- It has guaranteed continuity of the output surface.
- It is well suited to mathematical analysis.
- It can optimize the parameters of the output to improve the efficiency [59].

Disadvantages of the Sugeno Fuzzy method

- It is not intuitive.
- When using the higher order Sugeno method, it is complex.

3.4.4 Defuzzification

The reverse of fuzzification, defuzzification [52] converts the resulted fuzzy sets defined by the inference engine to the output of the model to a standard crisp signal[35]. This process gives output control signals to the controlled system [44]. There is no systematic procedure for choosing a good defuzzification strategy, but the selection of defuzzification procedure depends on the properties of the application[62].

There are several methods available for defuzzification of fuzzy control inference, these methods can be classified into different classes based on a common basis. [63]:

1. Maxima methods and derivatives
    1.1. Random choice of maxima (RCOM)
    1.2. First of maxima and last of maxima (FOM, LOM)
    1.3. Middle of maxima (MOM)
2. Distribution methods and derivatives
    2.1. General distribution methods
        2.1.1. Center of gravity (COG).
        2.1.2. Mean of maxima (MeOM).
        2.1.3. Basic defuzzification distributions (BADD).
        2.1.4. Generalized level set defuzzification (GLSD).
        2.1.5. Indexed center of gravity (ICOG).
        2.1.6. Semi-linear defuzzification (SLIDE).
    2.2. Specific distribution methods
        2.2.1. Fuzzy mean (FM).
        2.2.2. Weighted fuzzy mean (WFM).
        2.2.3. Quality method (QM).
        2.2.4. Extended quality method (EQM).

3. Area methods

3.1. Center of area (COA)

3.2. Extended center of area (ECOA)

4. Miscellaneous methods

4.1. Constraint decision defuzzi_cation (CDD)

4.2. Fuzzy clustering defuzzi_cation (FCD)

The most important ones for control are described in the following:

Center of gravity (COG):

It is the best known defuzzification operator method. It is a basic general defuzzication method that determines the value of the abscissa of the centre of gravity of the area below the membership function (Figure 3.18)

$$COG = \frac{\sum_{x_{min}}^{x_{max}} x.\mu(x)}{\sum_{x_{min}}^{x_{max}} \mu(x)}$$

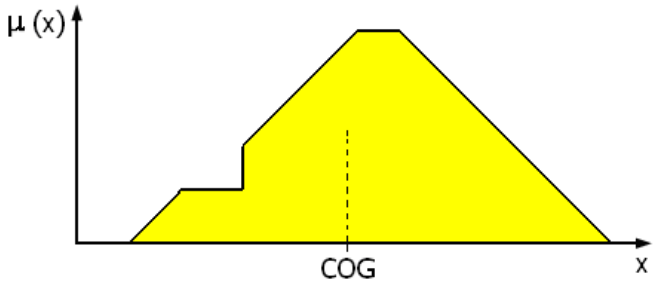


Figure 3-18 : Center of gravity (COG) defuzzification method

In general all defuzzification operators can be formulated in discrete form (via $\sum$) as well as in continuous form (via $\int$).

Middle of maxima (MOM):

This defuzzification method generates a crisp control action by averaging the support values which their membership values reach the maximum. If the core contains an odd number of elements, then the middle element of the core is selected. Otherwise, the defuzzification value depends on the implementation as follows:

For a crisp set S and $x_0 \in S$ let

$$s_{<x_0} = \{x \mid x \in s \quad and \quad x < x_0\} \quad and \quad s_{>x_0} = \{x \mid x \in s \quad and \quad x > x_0\}$$

For the membership A, values reach the maximum at core(A),

If |core(A)| is odd then

|core(A) < MOM(A)| = |core(A) > MOM(A)|

else

|core(A) < MOM(A)| = |core(A) > MOM(A)| ± 1

depending on the choice of implementation. (Figure 3.19)

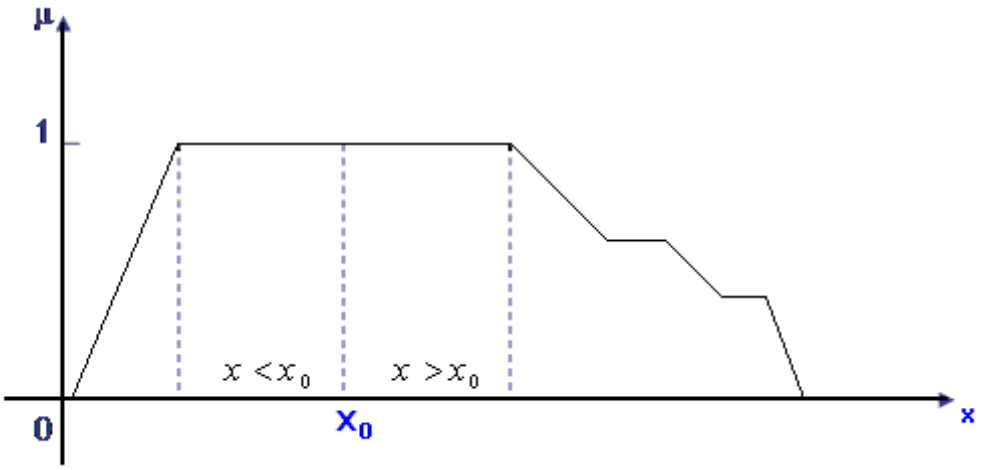


Figure 3-19 : Middle of maxima (MOM) defuzzification method

The weighted average method:

This method is used when the fuzzy control rules are the functions of their inputs. In general, the consequent part of the rule is: z = f(x,y) If W_i is the firing strength of the rule i, then the crisp value is given by:

$$Z = \frac{\sum_{i=1}^{n} W_i f(x_i, y_i)}{\sum_{i=1}^{n} W_i}$$

where n is the number of firing rules[64].

CHAPTER 4

Field Programmable Gate Arrays (FPGAs)

4.1 Introduction

There are two approaches for implementing today's digital control systems. The first approach is based on software. Programmable logic controller PLCs, Microcontrollers, microprocessors, and general purpose computers are tools for software implementation. The second approach is based on hardware. At latter time hardware implementation became achievable by means of digital logic gates and Medium Scale Integration components. When the system size and complexity increases, application-specific integrated circuit (ASIC) are utilized[65].

An ASIC is an integrated circuit (IC) customized for a particular use, rather than intended for general-purpose use. The ASIC must be fabricated on a manufacturing line, a process that takes several months, before it can be used or even tested. over the years the sizes have shrunk and design tools improved, the maximum complexity has grown from 5,000 gates to over 100 million. Modern ASICs often include entire 32-bit processors, memory blocks including ROM, RAM, EEPROM, Flash and other large building blocks. Designers of digital ASICs use a hardware description language (HDL), such as Verilog or VHDL, to describe the functionality. ASIC have some significant advantages because they are designed for a particular purpose they are very fast, efficient circuitry, and lower cost for high volume production. The disadvantages that it takes time for the ASIC vendor to manufacture and test the parts and the process of designing, testing and setting up fabrication facilities for the production of an ASIC is generally very expensive [66].
Programmable logic devices (PLDs) are configurable ICs and used to build reconfigurable digital circuits. Unlike a logic gate, which has a fixed function, a PLD has an undefined function at the time of manufacture, and before the PLD can be used in a circuit it must be programmed. The word "programmable" here mean hardware configuration of the chip, does not mean memory-processor architecture[65], and the PLD's function is defined by a user's program rather than by the manufacturer of the device.

There are many types of PLDs which differ in the architecture, density (system gates), and configuration technique. Common types of PLDs are programmable logic arrays (PLAs), complex programmable logic devices (CPLDs), and field programmable gate arrays (FPGA).

PLA is the first type of PLD family which had been appeared in the market in early 70's. PLAs were one-time programmable chips used to implement combinational logic circuits. It have two programmable planes provided any combination of “AND” and “OR” gates, as well as sharing of AND terms across multiple ORs, which can then be conditionally complemented to produce an output. A PLA device can be defined by a three parameters, number of inputs, number of AND gates (terms), number of OR gates (= number of outputs). A simplified diagram of the PLA device with 4 inputs, 4 terms and 3 outputs is shown in Figure 4.1. Note that logic gates are fixed, only the programmable planes are programmable based on fuses that are programmed by burning[67]. This layout was very flexible, but relatively slow, since the propagation delay time was high[68].

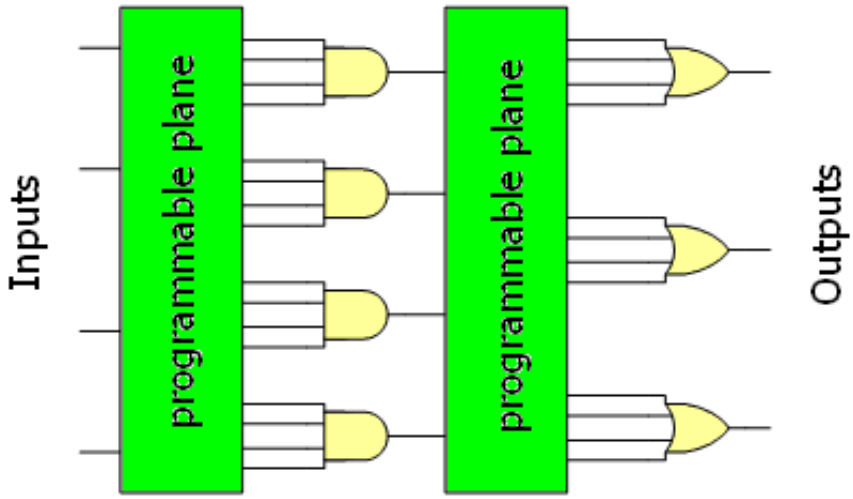


Figure 4-1: PLA constructions .

There was also a variation of this architecture in which only the first programmable plane (before AND gates) was programmable, and the second one was hardwired. These devices were called Programmable Array Logic (PAL) [67]. It was faster and less complex software, but without the flexibility of the PLA. This new architecture shown in Figure 4.2 have arrays of transistor cells arranged in a "fixed-OR, programmable-AND" plane used to implement "sum-of-products" binary logic equations for each of the outputs. These category of PLD (PLA and PAL) devices are often called Simple PLD or SPLD.

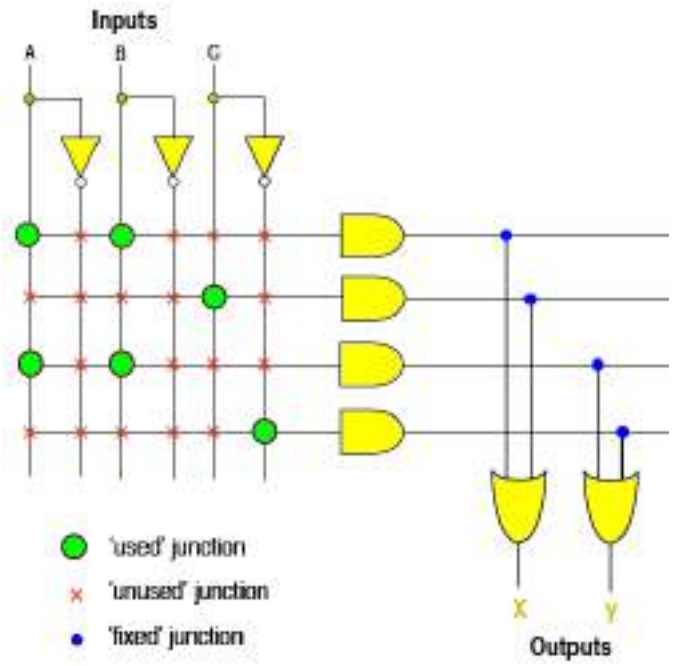


Figure 4-2 : PAL architecture .

The architecture had horizontal and vertical interconnect tracks. There are a fuse at each junction as shown in Figure 4.3. The programmable elements (shown as a fuse) connect both the true and complemented inputs to the AND gates. These AND gates, also known as product terms, are ORed together to form a sum-of-products logic array. With the aid of software tools, and device programmer designers blowing all unwanted fuses to select which junctions would not be connected[69].

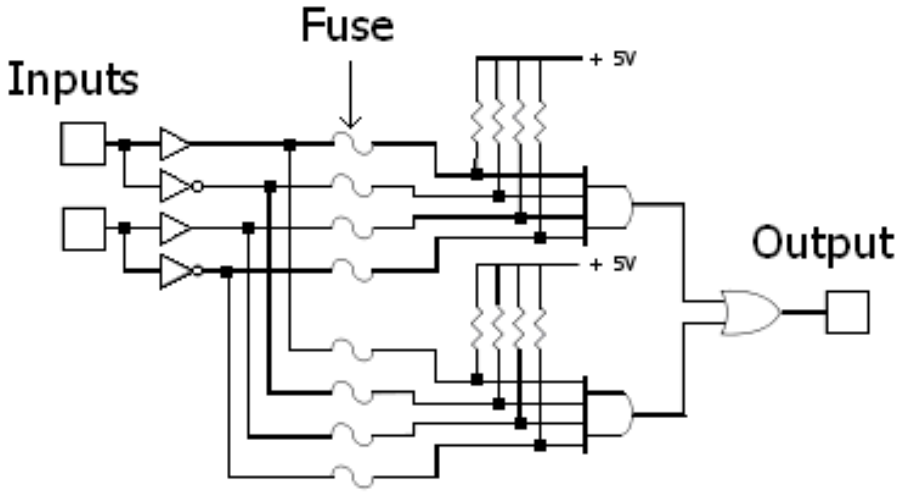


Figure 4-3 : Simplified programmable logic device.

Figure 4.4 show another way to extend the density of the simple PLDs. It is a Complex Programmable Logic Devices (CPLD). The concept is to have a few logic blocks or macrocells in the borders of the chip, and a connection matrix located at the central part. Each macrocell has a structure similar to PLA. So, a CPLD device can be also seen as a set of PLAs on one chip with programmable interconnects. Simple logic paths can be implemented within a single block. More sophisticated logic will require multiple blocks and use the general purpose interconnect in between to make these connections. CPLDs are usually flash-based, that is, the configuration of macrocells and the interconnection matrix is defined by contents of the on-chip flash memory. It means that CPLD need not to be configured after each power-up[67].

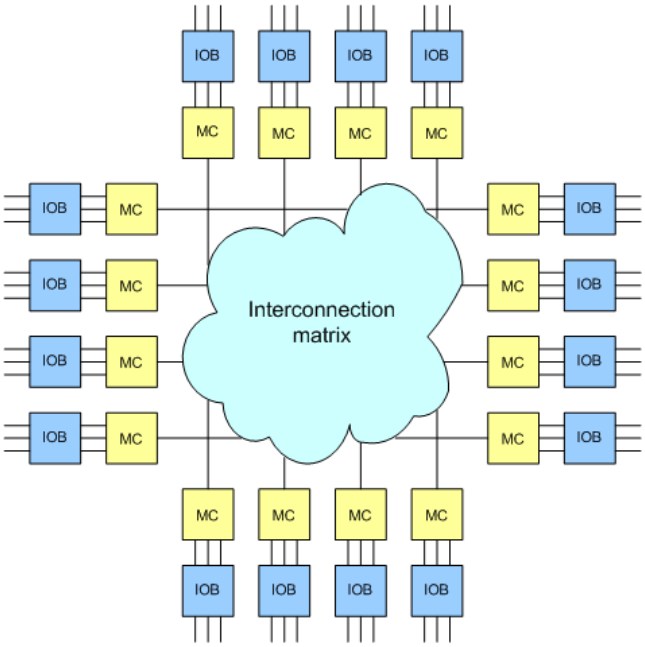


Figure 4-4: CPLD architecture.

PLD technology has moved on from the early days with companies such as Xilinx producing ultra-low-power CMOS devices based on flash memory technology. Flash PLDs provide the ability to program the devices time and time again, electrically programming and erasing the device[70]. Xilinx company introduced a new concept in 1985, it was to combine the user control and time to market of PLDs with the densities and cost benefits of gate arrays. Then the FPGA was found[68].

Field programmable gate arrays (FPGAs) are a special class of ASICs which differ from mask-programmed gate arrays in that their programming is done by end-users at their site with no IC masking steps. An FPGA is an integrated circuit consists of an array of logic blocks that can be configured after it is manufactured by the user in order to implement digital logic functions of varying complexities[74]. The complexity of a logic circuit depends on the functions to be implemented. As the functions in an application grows the complexity of the design increases. Before the advent of the programmable design, the logic circuits were built using the standard logic. The complexity of the circuit will increase as the number of gates increase[73]. An FPGA is under your complete control, this means that you can design, program, and make changes to your circuit whenever you wish[68].

4.2 FPGAs Architecture

FPGAs come in a wide variety of sizes and with many different combinations of internal and external features. What they have in common is that they are composed of relatively small blocks of programmable logic as shown in Figure 4.5[68]. These blocks, each of which typically contains few registers and few dozen low-level, configurable logic elements, are arranged in a grid and tied together using programmable interconnections[74], show Figure 4.6 [75].

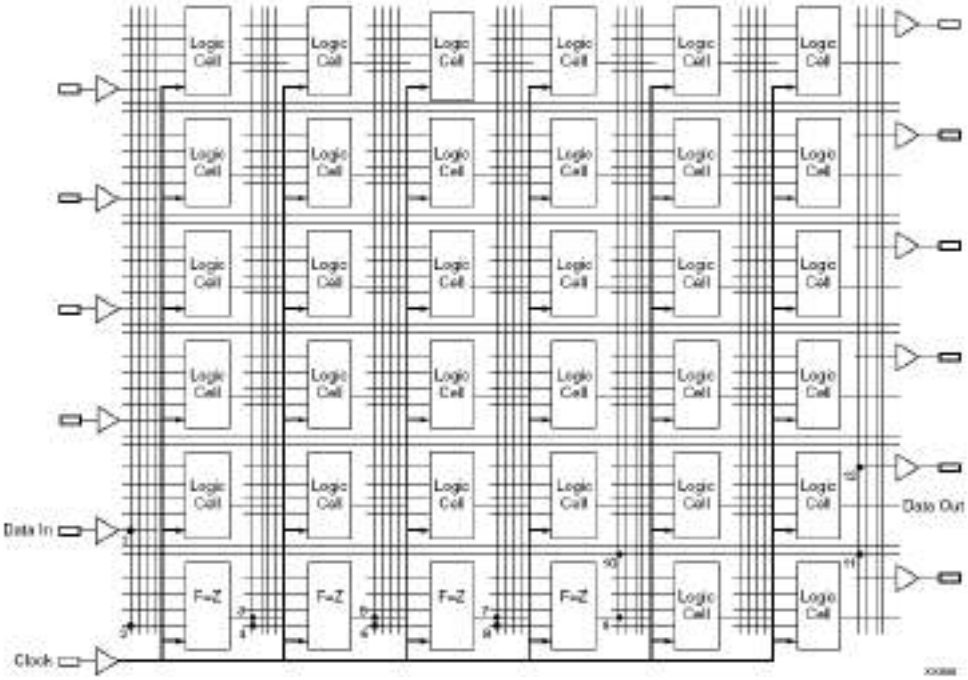


Figure4-5: FPGA Architecture.

By having programmable interconnections and configurable logic elements, the system can be configured to mimic any combination of logic functions as long as the overall design can be fitted into the available number of logic elements and programmable interconnections.

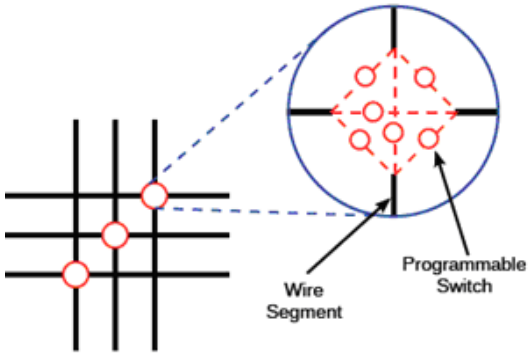


Figure 4-6 : Routing Switch Box .

As shown in Figure 4.7[76], the logic blocks that make up the bulk of the device are based on Look-Up Table (LUT) combined with one or two single-bit registers (flip-flop) and additional logic elements such as clock enables and multiplexers, see Figure 4.8. These basic structures may be replicated many thousands of times to create a large programmable hardware fabric.

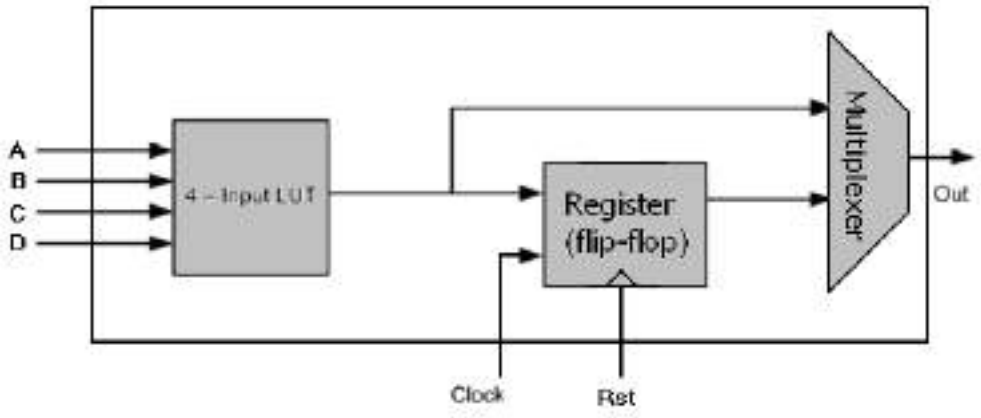


Figure 4-7 : FPGA logic element.

In more complex FPGAs these general-purpose logic blocks are combined with higher-level arithmetic and control structures (eg. adders), in support of common types of applications such as signal processing. In addition, specialized logic blocks are found at the periphery of the devices that provide programmable input and output capabilities. Current FPGAs offer complexity equivalent to a million gate conventional gate array and typical system clock

speeds of hundreds of MHz[72]. The largest FPGA manufacturer, Xilinx, produces chips that contain 758,784 logic cells, 3.2 MB of on-chip Block RAM, 864 embedded DSP blocks, and are capable of running at 600 MHz [77].

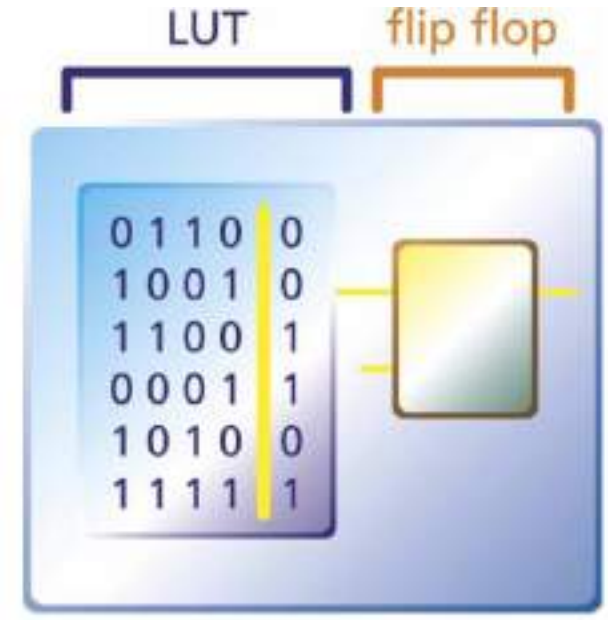


Figure 4-8: **SRAM Logic Cell**

4.3 FPGA Programming Technologies:

FPGA programming technologies range from one-time-programmable elements (such as those found in devices from Actel and Quicklogic) to electrically erasable or SRAM-based devices such as those available from Altera, Lattice Semiconductor, and Xilinx.

A program written by someone defines FPGA's function, the program is loaded from an external memory each time the device is powered up. This user programmability gives the user access to complex integrated designs without the high engineering costs associated with application-specific integrated circuits.

For SRAM-based and electrically erasable FPGAs, if it is programmed during system manufacturing to perform one specific task, it can be reprogrammed while the product is in the field or upgrading may be as simple as providing an updated Flash memory card or obtaining a new binary device image from a website or CD-ROM.

Hardware applications implemented in FPGAs are generally slower and consume more power than the same applications implemented in custom ASICs. But The main advantage of FPGAs over mask-programmed ASICs is the fast turnaround that can significantly reduce design risk because a design error can be quickly and inexpensively corrected by reprogramming the FPGA, with lowered risk and cost[74].

Individually defining the many-switch connections and cell logic functions would be a daunting task. Fortunately, this task is handled by special software. Xilinx offers complete powerful and flexible software that enable the implementation of designs in Xilinx PLDs, such as WebPACK ISE software. Designs can be described easily and quickly using a description language such as VHDL, Verilog, or with a schematic capture package.
The software translates a user's schematic diagrams or textual hardware description language code and then places and routes the translated design. Libraries of more complex function macros such as multipliers and counters providing common circuits that are already optimized for speed or area to simplify the design process[78].

FPGA process compiling an abstract hardware design from Hardware Design Language (HDL) code to an FPGA programming file, it is very time consuming compared to software compilation[79], because the basic work flow for compiling and testing an FPGA design is take several stages as shown in Figure 4.9. These stages are[72]:

- Logic design and simulation.
- Placement, routing and connectivity check.
- Programming.

After the Design Entry stage, the design can be synthesised, a process that involves conversion of an HDL description (program or a schematic) to a so-called netlist. The netlists contain the structural description of the design and are used for functional simulation. Synthesis is performed by a special software called synthesizer. The netlist is mapped onto particular device's internal structure in a process that is called implementation. The main phase of the implementation stage is place and route or layout, which allocates FPGA resources (such as logic cells and connection wires). Then these configuration data are written to a special file by a program called bitstream generator.

This is followed by accurate timing analysis. During this analysis special software checks whether the implemented design satisfies timing constraints specified by the user such as clock frequency. The final procedure is a bitstream file, which can be directly downloaded into the targeted device via the usp, serial or parallel interfaces of a PC.

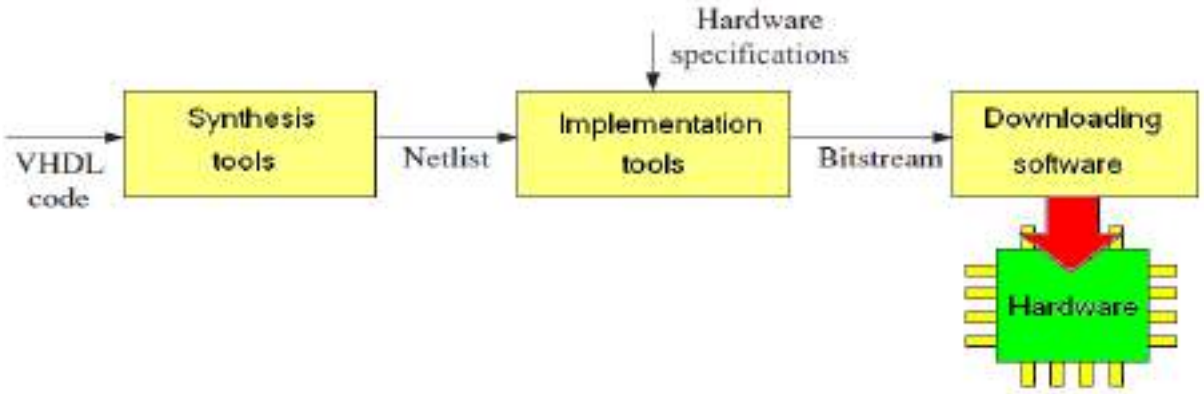


Figure 4-9: Simplified block diagram of the hardware design process.

4.4 Fuzzy Controller Design using FPGA

The implementation of a fuzzy controller using FPGA generally requires a large number of logic gates. It is consists of five steps as shown in Figure 4.10. The basic step is to collect the information on the number of inputs and outputs to choosing an optimum FPGA. An ADC/DAC would be required to map from analog to digital or vice-versa, since the plant to be controlled works on an analog domain and the FPGA works on a digital domain. The second step is to design the fuzzy sets using the VHDL Language. The design basically depends on the operator's experience[80]. The third step is to write the rules base. These rules basically instruct the action to be performed for various combinations of input. The fourth step is to design the output fuzzy set. The last step is the testing step where the controller is basically tested by connecting it to the plant, which has to be controlled.

The membership functions, rules and the scaling factors are changed each time in order to attain a perfect process control design. Since this is a repetitive process, there should be an arrangement which can help the user to simulate his process model in a software environment and then design the same model on hardware.

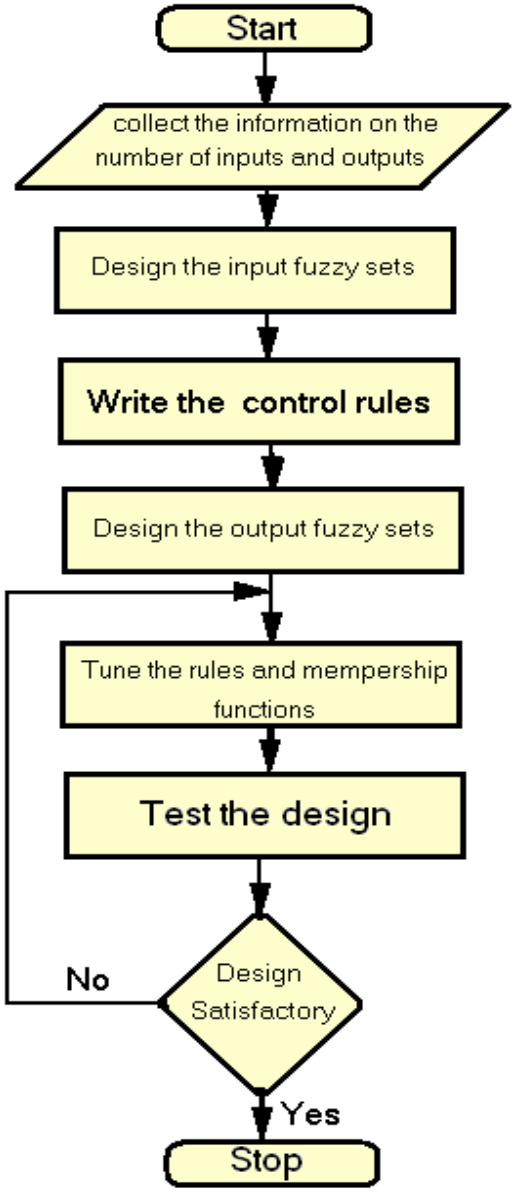


Figure 4-10: **Design steps for an FLC using an FPGA .**

4.5 Xilinx Spartan-3AN FPGA Starter Kit:

High volume applications require efficient and accurate FPGA evaluation with rapid access to the device features, and easy-to-use development tools. So Xilinx 700K-gate XC3S700AN Spartan-3AN nonvolatile FPGA (Figure 4.11) is a complete, low-cost solution for quick evaluation. The Starter Kit comes complete with an evaluation board, power supply, design tools, reference designs, and accessories. Xilinx provide this product to implement Spartan-

3AN FPGA designs in the shortest possible time with an evaluation board, free design tools and reference designs all in one starter kit[81].

Figure 4-11: Spartan-3AN Starter Kit Board .

Spartan-3AN board benefits such as low cost, power savings, system flexibility with the industry's largest on-chip Flash make it makes it an ideal replacement for fixed-logic gate arrays and for ASICs. Depending on that and after features checking for the needs of my project in this thesis, I decided to select Xilinx 700K-gate XC3S700 AN Spartan-3AN nonvolatile FPGA Starter Kit Board. Also it's memorable to mention that this board is very modern due to its manufacturing date (November 19, 2009). The key Components features of the Spartan-3AN Starter Kit are available at the **Appendix A**.

CHAPTER 5

Designing Sun Tracker System Using FLC on FPGA

5.1 Overall System Design and Implementation

This research presents the process used to design two different fuzzy controllers for the solar energy system using FPGA technique. The block diagram in Figure 5.1 explains the concept of this system. The system has two controllers, one to control the PV panel position to be aligned to the sun all the day, and another controller for tracking the maximum power point, that to increase the efficiency of PV panel.

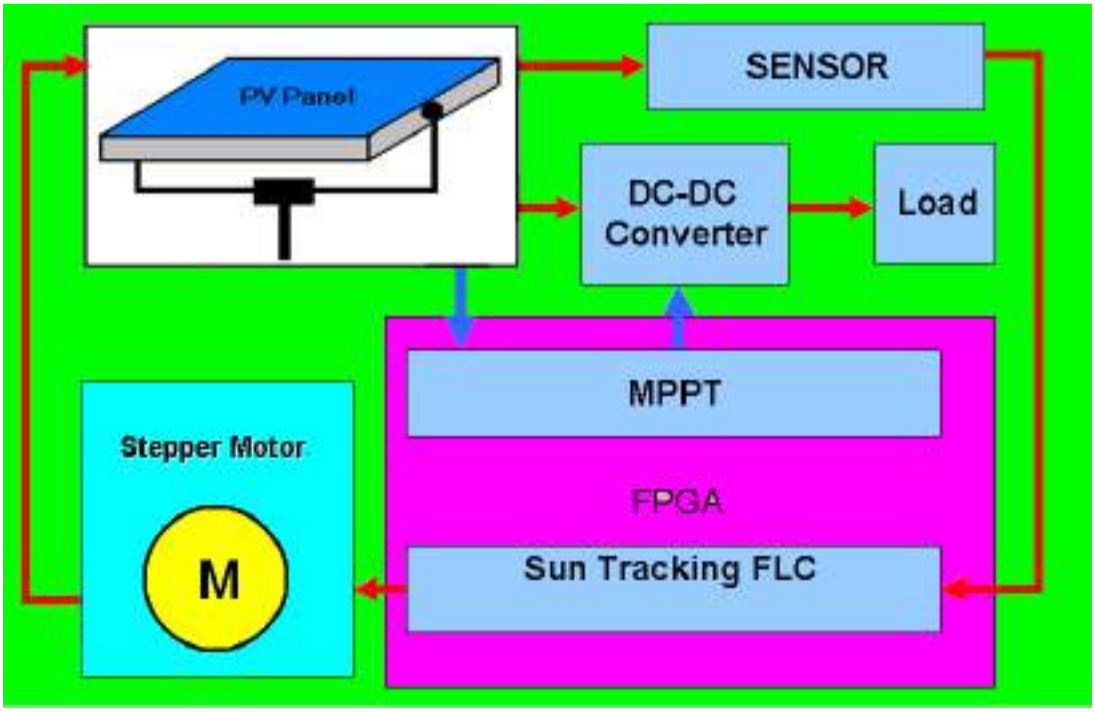


Figure 5-1: Block diagram for the overall control system.

5.2 Sun Tracker

Solar tracking system uses a stepper motor as the drive source to rotate the solar panel (see Figure 5.2). The position of the sun is determined by using a tracking sensor, the sensor reading is converted from analog to digital signal, then it passed to a fuzzy logic controller implemented on FPGA. The controller output is connected to the driver of the stepper motor to rotate PV panel in one axis until it faces the sun.

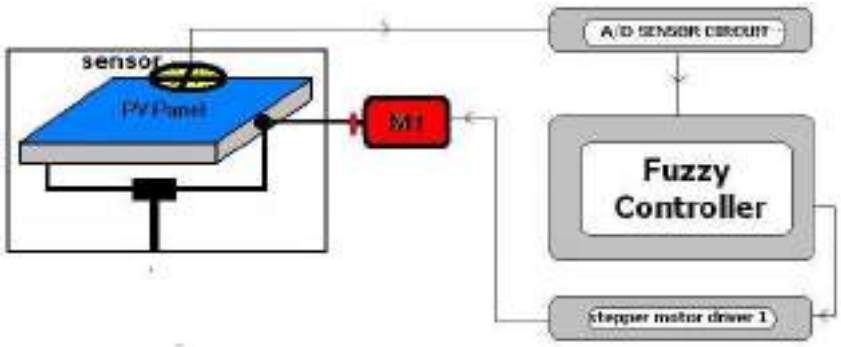


Figure 5-2 : Block diagram for the sun tracker system .

5.3 Sensors:

There are two sensors used in the system: photo sensor, and position sensor.

5.3.1 Photo Sensor:

Light dependent resistor (LDR) is used to construct the sensor, because it is the most reliable sensor that can be used for light sensing. LDR is basically a resistor whose resistance varies with intensity of light, so more intensity gives less resistance. Different LDR sensors available in the market are shown in Figure 5.3, the biggest size is used to construct the sensor because the more area of the sensor mean more its sensitivity or less time taken for output to change when input changes.

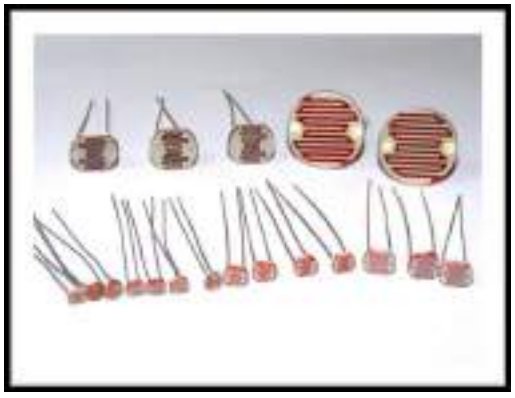


Figure 5-3: Different LDR sensors.

5.3.2 Tracking Sensor Design

To sense the position of Sun in one axes say east/west, two LDR sensors are mounted on the solar panel and placed in an enclosure as in Figure 5.4. It has a response which is similar to the human eye. The east and west LDR sensors compare the intensity of received light in the east and west. The right side of Figure 5.4 describes the case when Sun's position shifts, here

the light source intensity received by the sensors are different, the system obtains signals from the sensors' output voltage in the two orientations. The system then determines which sensor received more intensive light based on the sensor output voltage value interpreted by voltage type A/D converter. The system drives the step motor towards the orientation of this sensor. If the output values of the two sensors are equal, as the left side of Figure 5.4, the output difference is zero and the motor's drive voltage is zero, which means the system has tracked the current position of the sun.

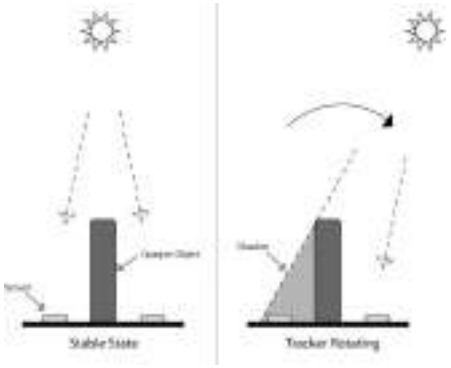


Figure 5-4: how sensor work.

The tracking sensor is composed of two similar LDR sensors, which are located at the east, west, or south, and north to detect the light source intensity. The LDR sensor forms a 45° angle with the light source. At the LDR sensor positions, brackets isolate the light from other orientations to achieve a wide-angle search and quickly determine the sun's position, see Figure 5.5.

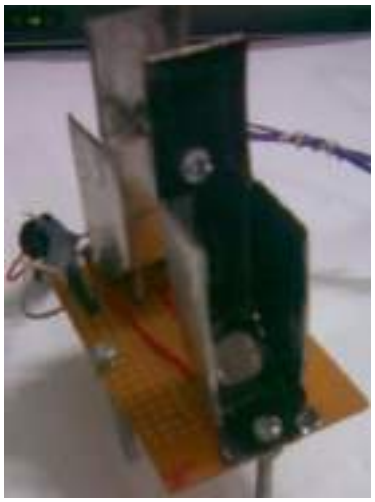


Figure 5-5: Tracking Sensor Internal Design

5.3.3 LDR Connection:

The basic circuit for connecting an LDR is shown in the Figure 5.6. It can be connected in a simple potential divider to give a voltage output buffered by a unity gain amplifier, the IC LM324 is used. Vo is fed into the microcontroller through analogue to digital conversion. When under bright light, the resistance of the LDR is very low and vice versa. So the Vo input voltage to the PIC will be small for the LDR exposed to lesser intensity.

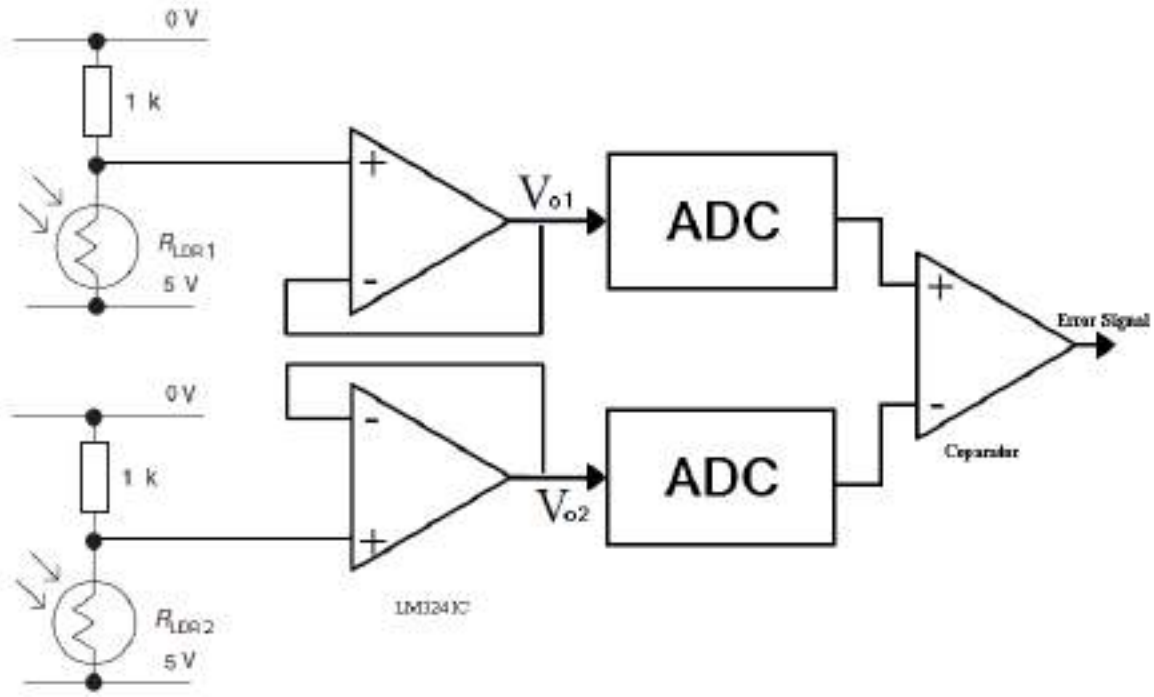


Figure 5-6 : LDR connection.

5.3.4 Position Sensor:

Position sensor used to determine the location of the PV panel to prevent the panel from the impact when it reach the edges, and to get the PV panel to the starting point at the night. This sensor used a variable resistor (potentiometer) located on the rotor of the motor and rotate with it, and the value of the resistor (R) varies with the rotation as shown in Figure 5.7. When the position sensor reach the values at the PV at the edges, the controller stopped the motor and immune it from rotating in that direction. At the night the LDRs sensors are very dark light and their values are very big, in this situation the controller go to night subroutine to rotate the PV panel until the position sensor has the starting point value.

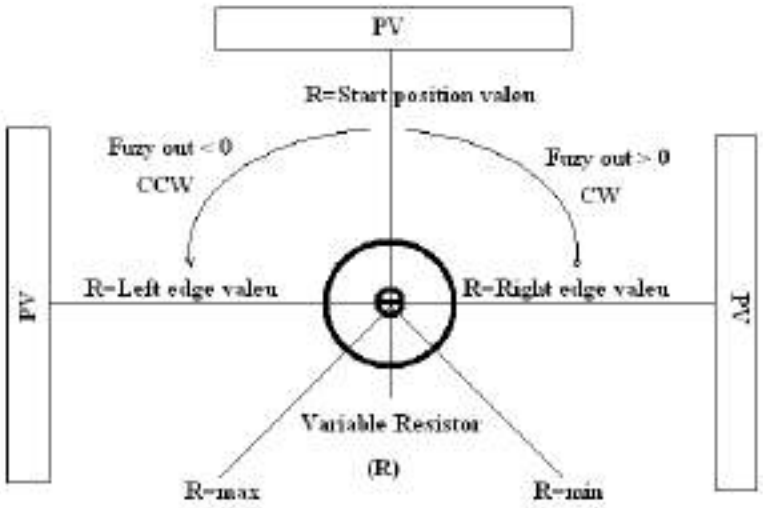


Figure 5-7: **Position sensor.**

Figure 5.8 show the algorithm of extracting the motor control signals depending on sensors reading and the output of the controller. Were R is the value of the position sensor, and En is the enable signal to rotate the motor.

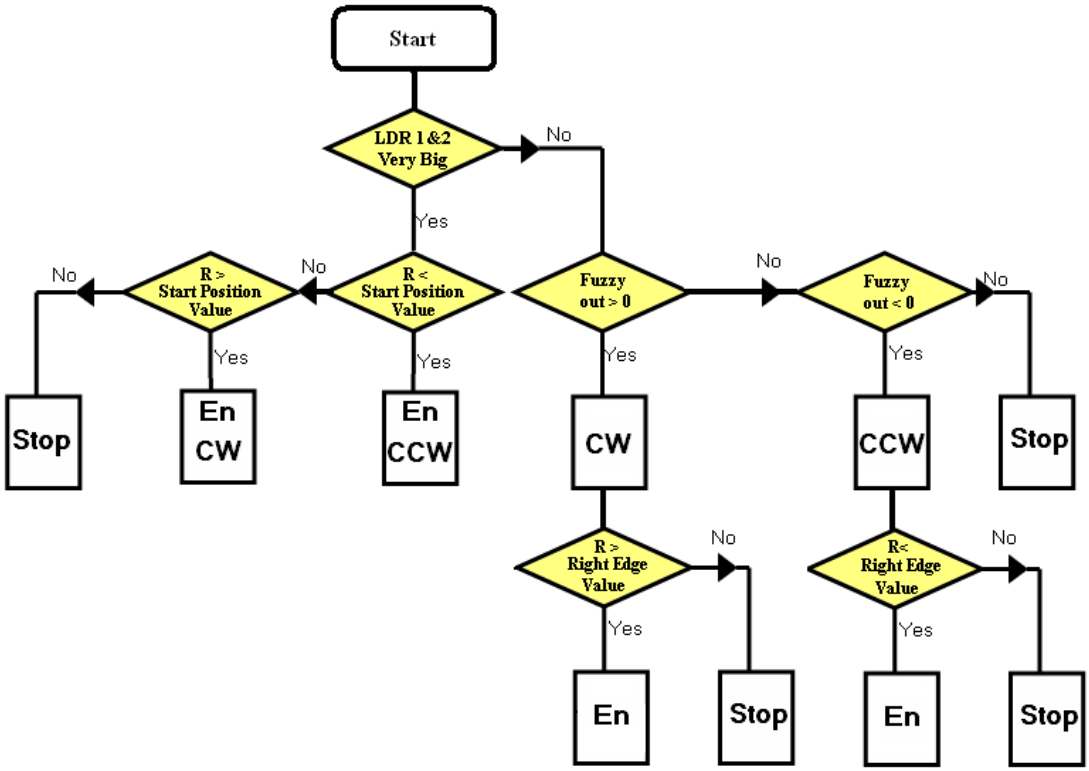


Figure 5-8: **Motor Control Signals algorithm.**

5.3.5 Analog to Digital Converter (ADC)

The PIC16F877A single chip microcontroller is used as ADC. This microcontroller contains 8 kbytes of program memory, a 256 Bytes of temporary data RAM and 1 kbytes of EEPROM. It also contains 8 multiplexed analog channels, a 10 bits analog converter and a PWM generator module. These features make the PIC16F877A a useful and powerful single chip microcontroller in converting the analog sensors reading to digital data . Figure 5.9 shows the ADC conversion characteristic, where the input voltage is represented on the horizontal axis and digital output on the vertical.

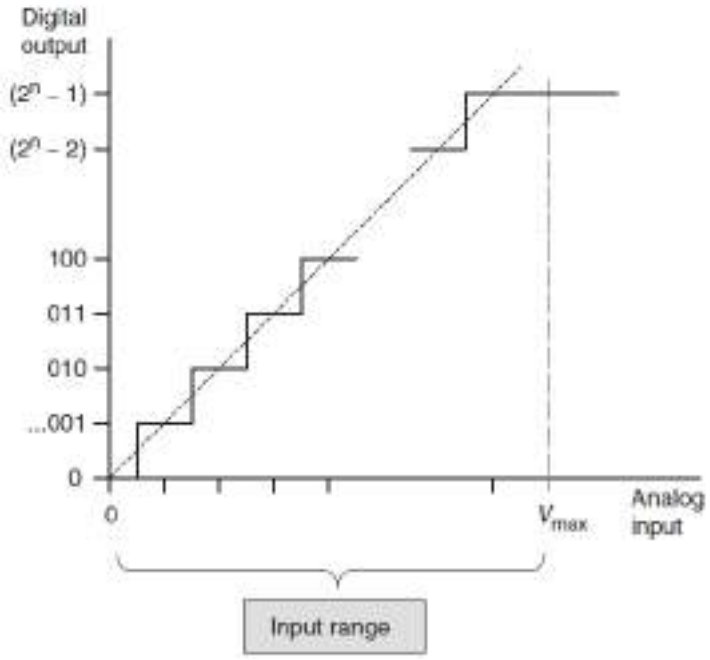


Figure 5-9:**The ideal ADC input/output characteristic**

For an n-bit ADC, the maximum output value will be $(2^n - 1)$. For an 8-bit ADC, the final value will be $(2^8 - 1)$, or 11111111(binary), or 255(decimal). The input range starts from zero and goes up to the value V_{max} =5V. The horizontal axis is divided into exactly 2^n equal segments, each centered on an output transition. It can be seen intuitively from the diagram that the more the number of output bits, the more will be the number of output steps and the increasing in the resolution. Therefore, it has a resolution of $(Vr/2^n)$=(5/255)=(0.0196), where Vr is the input voltage range.

5.4 Fuzzy Logic Controller:

FLC has been constructed and the block diagram in Figure 5.10 shows the FLC for the sun tracker system.

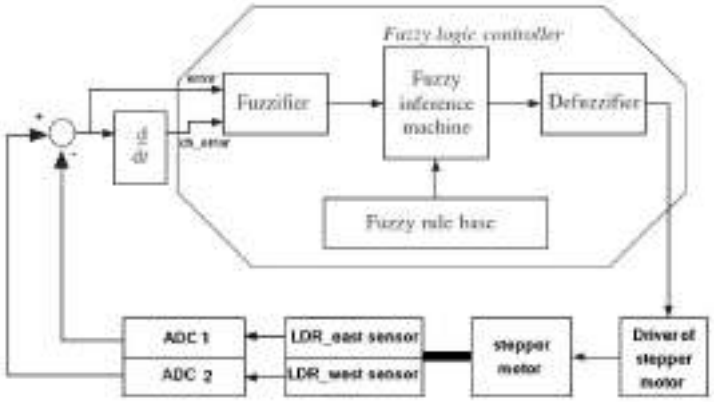


Figure 5-10: FLC controller for the sun tracker system.

5.4.1 FLC design

FLC has two inputs which are: error and the change in error, and one output feeding to the stepper motor driver. The phasor plot in Figure 5.11 explained a method used in reaching the desired degree value at the equilibrium point to satisfy the stability in the system. For example, at stage A the error is positive (desired degree –actual degree) and the change error (error – last error) is negative which means that the response is going in the right direction; hence, the FLC will go forward in this direction. Using the same criteria at stage B, the error is negative and CE is bigger negative; hence, the response is going in wrong direction so FLC will change its direction to enter Stage C, until reaching the desired degree.

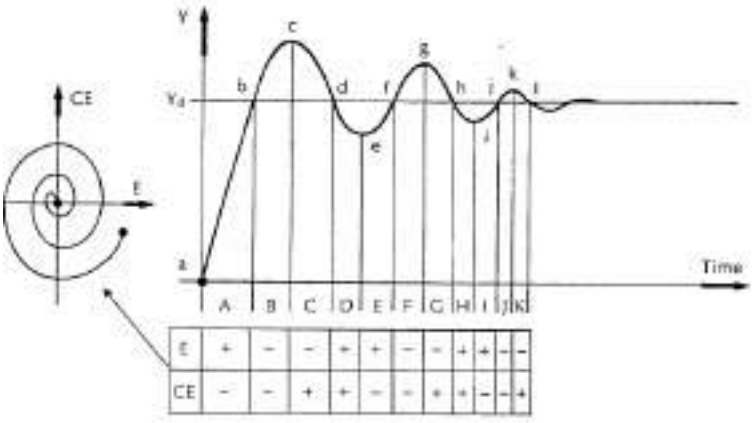


Figure 5-11: Error and change in error approach in FLC

There are two widely used approaches in FLC implementation: Mamdani and Sugeno. In this thesis, Mamdani approach has been used to implement FLC for the sun tracker. FLC contains three basic parts: Fuzzification, Base rule, and Defuzzification.

- **Fuzzification**

Figure 5.12 illustrates the fuzzy set of the Error input which contains 7 Triangular memberships

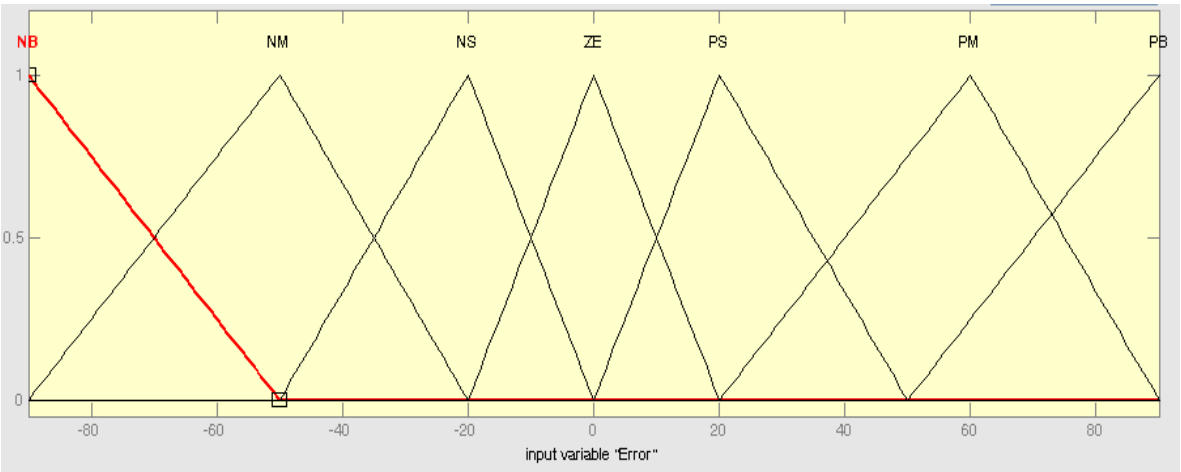


Figure 5-12: Error fuzzy set of FLC.

Figure 5.13 illustrates the fuzzy set of the Change of Error input which contains 7 Triangular memberships.

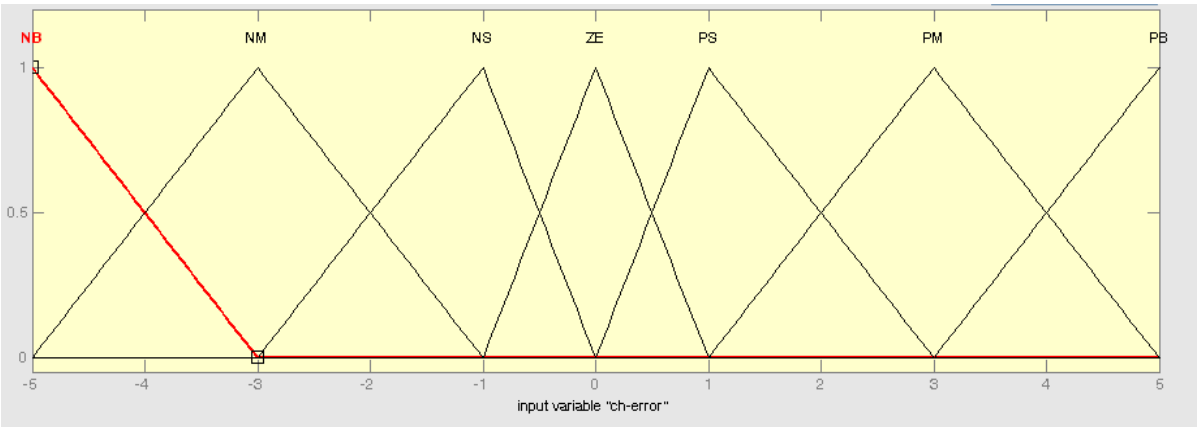


Figure 5-13: change in error fuzzy set of FLC.

Figure 5.14 illustrates the fuzzy set of the output which contains 7 Triangular memberships.

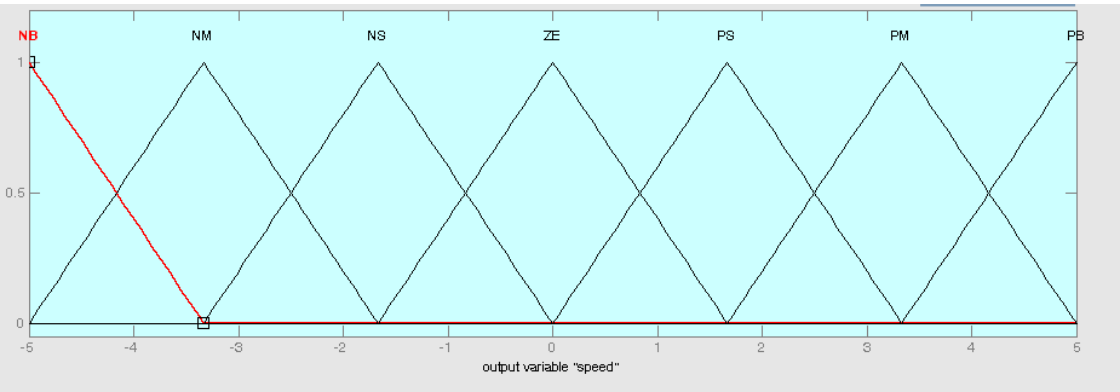


Figure 5-14: Fuzzy set of FLC output entering to stepper motor driver.

- **Control rule base**

The knowledge base defining the rules for the desired relationship between the input and output variables in terms of the membership functions illustrated in Table 5.1. The control rules are evaluated by an inference mechanism, and represented as a set of:

IF Error is ... and Change of Error is ... THEN the output will … .

For example: Rule1: IF Error is NS and Change of Error is ZE THEN the output is NS.

The linguistic variables used are:

NB: Negative Big.

NM: Negative Medium.

NS: Negative Small.

ZE: Zero.

PS: Positive Small.

PM: Positive Medium.

PB: Positive Big.

Table 0-1: Control rule base for fuzzy controller.

Er \ CE	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZE
NM	NB	NB	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PB	PB
PB	ZE	PS	PM	PB	PB	PB	PB

Figure 5.15 shown the surface of the base rules using in FLC which is the representation for the inputs and output values of the controller in three dimentions.

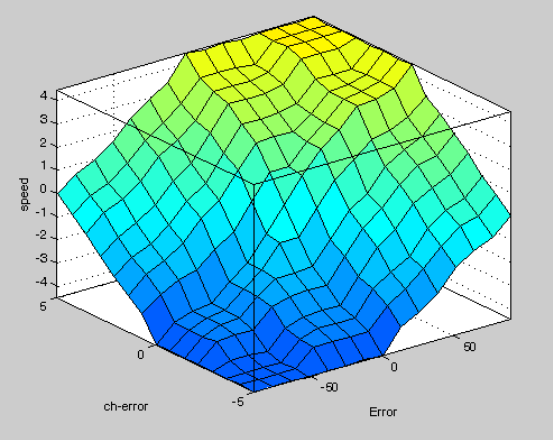


Figure 5-15 : Rule surface of FLC.

- **Defuzzification**

As illustrated in Chapter 3, the center of gravity method is widely used in Mamdani approach which has been selected in this thesis to compute the output of the FLC, which is the motor speed as:

$$Speed = \frac{\sum_{i=1}^{n} S_i * \mu(S_i)}{\sum_{i=1}^{n} \mu(S_i)} \qquad \textbf{(5.1)}$$

5.4.2 Fuzzy Logic Controller Simulation on Matlab/Simulink:

Figure 5.16 illustrates the Simulink block diagram for the Fuzzy controller for sun tracker system.

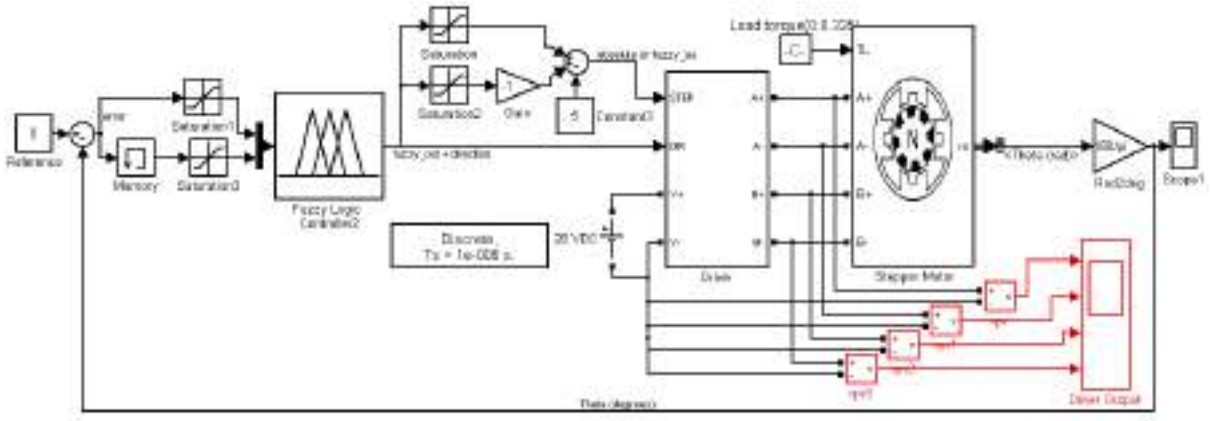


Figure 5-16: Testing the FLC in the sun tracker system using Matlab/Simulink.

The controller has been tested using Simulink motor module in MATLAB, by applying the step input and initial degree of the rotor is -10 degree. The output step response is shown in Figures 5.17. The range from -10 to 0 degree takes 5 steps since each step in our motor is 1.8 degree, so (10/1.8)= 5 steps.

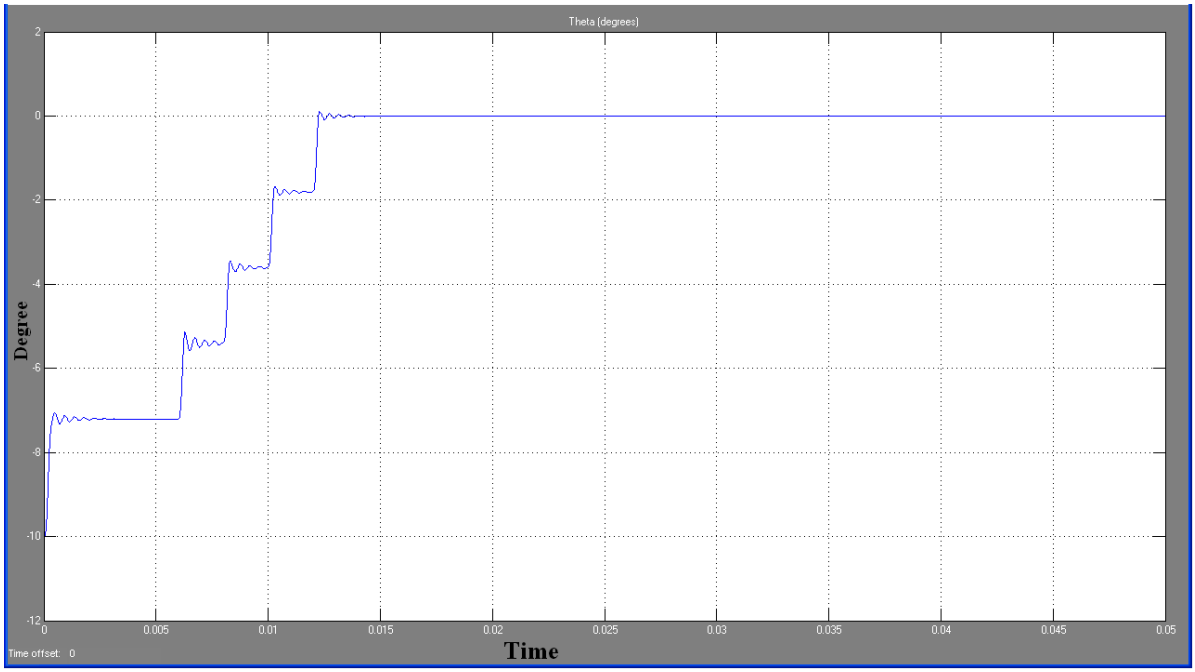


Figure 5-17: Output Degree.

Figure 5.18 is a zoom for one motor step, the overshoot is 1.3%, error steady state is 0.005 degree, and the settling time is 0.016 second.

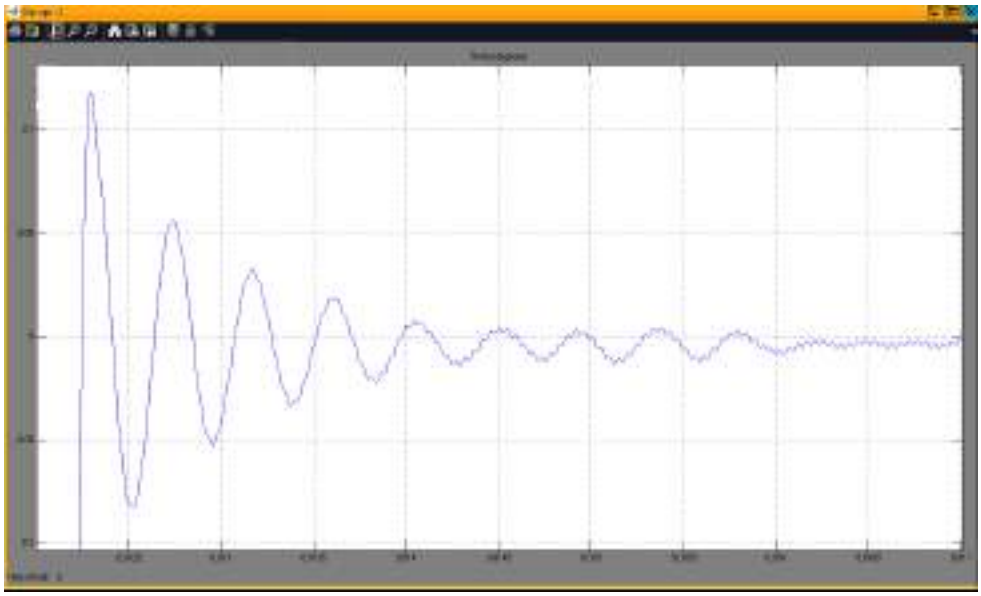


Figure 5-18: A zoom for one motor step.

5.5 Stepper Motor Driver:

Bipolar Stepper Motor is used in the sun tracker system, to rotate the PV panel to face the sun. It has a 1.8-degree for each step, and four wires connects four coils as shown in Figure 5.19.

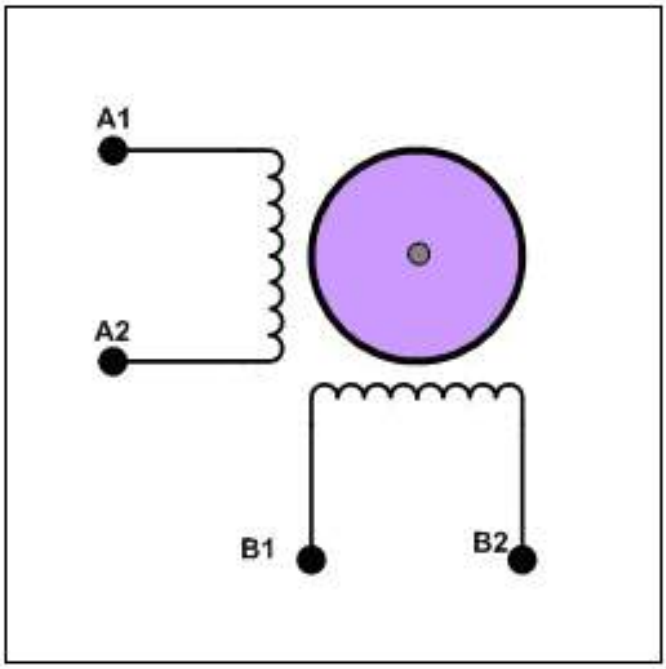


Figure 5-19 : Bipolar Stepper Motor.

Stepper motors require more power than other components in the circuit, so they are connected with a separate power supply. The voltage is applied to each of the coils in a sequence as shown in table 5.2 to control the stepper.

Table 0-2: **Full-step phase sequence**

Step	A1	A2	B1	B2
1	1	0	1	0
2	1	0	0	1
3	0	1	0	1
4	0	1	1	0

When the sequence is applied as steps from 1 to 4, the motor will rotate in clockwise direction, and rotate in the other direction if the steps is in reverse order. Figure 5.20 shows the output of the driver in the last Simulink (Figure 5.16).

Figure 5-20: The output of the driver

To use the stepper motor, a driver circuit used to obtain the stepping sequence as shown in Table 5.2. The driver is shown in Figure 5.21.

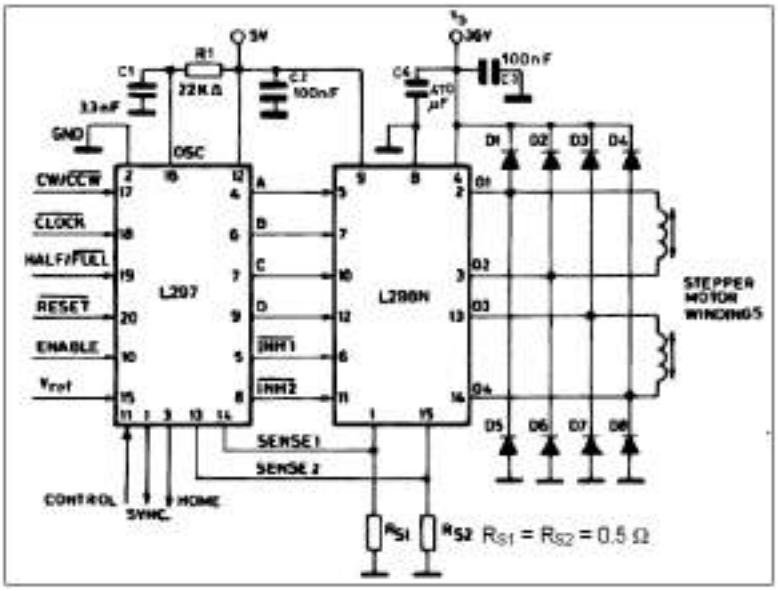


Figure 5-21: Stepper motor driver as block diagram.

The main components in the driver are:

1- Stepper motor controller (L297 IC):

The L297 Stepper Motor Controller IC, here is used to generates four phase drive signals for four phase bipolar step motor. It receive 5 control signals from the controller, these signals are:

- CLOCK: An active low pulse on this input advances the motor one increment. The step occurs on the rising edge of this signal. Its frequency controls the speed of the motor. The output of the FLC is applied to an interface block subroutine in the FPGA and converted to a clock wave have a frequency appropriate with its value.
- CW/CCW: Clockwise/counterclockwise direction control input.
- HALF/FULL: Half/full step select input. When high selects half step operation, when low selects full step operation.
- RESET: An active low pulse on this input restores the translator to the home position (state 1, ABCD = 0101).
- ENABLE: When low (inactive), A, B, C and D are brought low.

2- Dual full-bridge driver (L298 IC)

The L298 is an integrated circuit. It is a high voltage, high current dual full-bridge driver designed to drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. An additional supply input is provided so that the logic works at a lower voltage. In Appendix B, there is more information about this IC. Figure 5.22 shows the stepper motor driver PCB kit.

Figure 5-22: Stepper motor driver PCB kit.

5.6 Implementing Fuzzy Logic Controller on an FPGA:

The fuzzy logic controller designed earlier is implemented on Xilinx XC3S700AN FPGA card as shown in Figure 5.23.

Figure 5-23: FLC on FPGA card.

The program is written by VHDL language using Xilinx_ISE 11.1 software program (Figure 5.24). The FLC and LCD display screen subroutine implemented by many VHDL files, then these files are linked together by one top module. Then, the user constraints file (UCF) is implemented, this file is used to define the input and output ports on the card. The main difficulties faced in programming the FPGA were in programming the mathematics operations like product and division.

Figure 5-24: **VHDL language in Xilinx_ISE 11.1 software program.**

Figure 5.25 shows the RTL schematic diagram in Xilinx software RTL Viewer to view a schematic representation for the FLC and other components after implementing it on Xilinx_ISE 11.1 software.

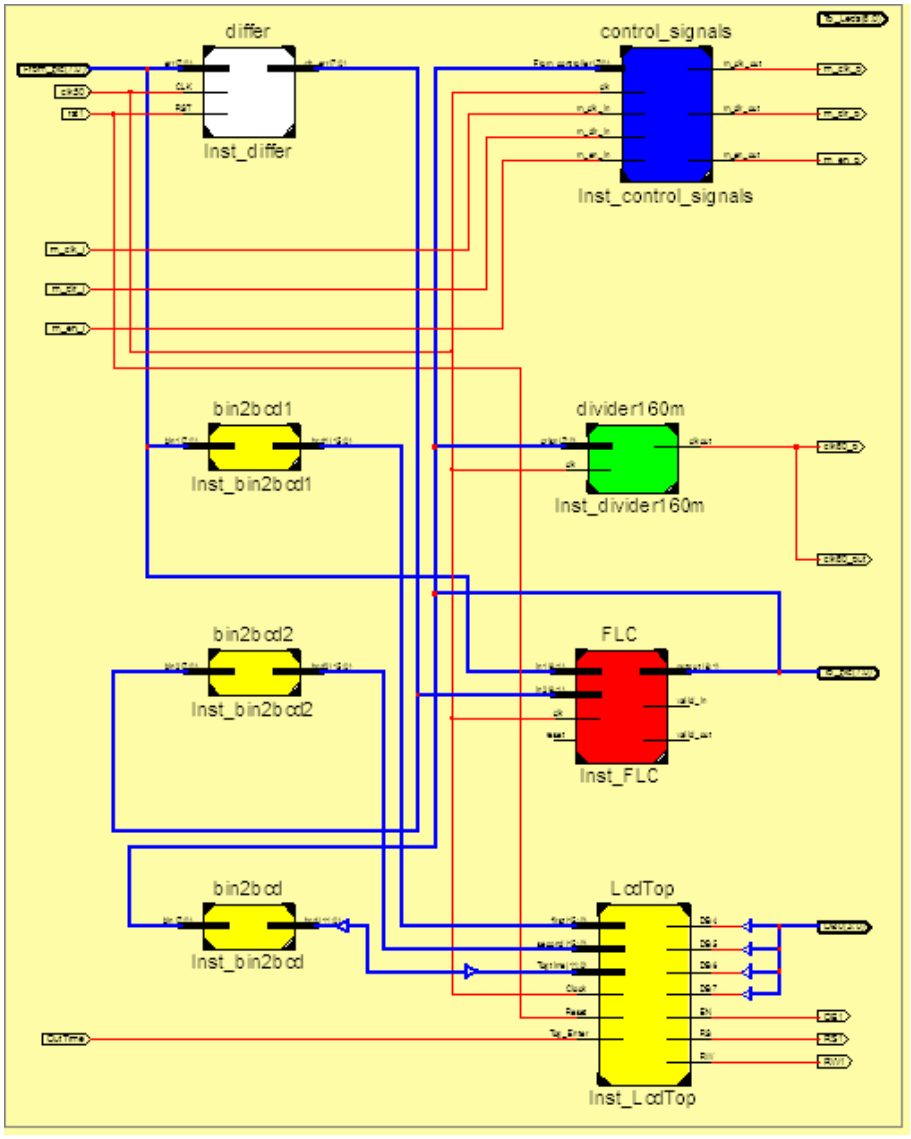


Figure 5-25: RTL schematic diagram for the FLC with other blocks.

The fuzzy logic controller appears as a red block labeled by name 'FLC', the first input of the controller is error signal takes before differentiator (white block), and other input of the controller is change in error signal takes from after differentiator. The output of the controller is passed through three blocks, the first block which have green color to convert the crisp value to clock wave have a frequency appropriate with this value to control the speed of the motor, the second block which have blue color to extract the other motor control signal such

the direction of the motor and the rotation enable signal as explained in section (5.2.2.4), the third block which have yellow color (LcdTop block) is for LCD display screen to display the output of the controller. Other blocks are input/output data transfer.

In order to transform a behavioral level VHDL design into a hardware design, a synthesis and implementation processes must be made as explained in Chapter 4. Double click on the Synthesis and on the Implement design process in Xilinx_ISE 11.1 software as shown in Figure 5.26.

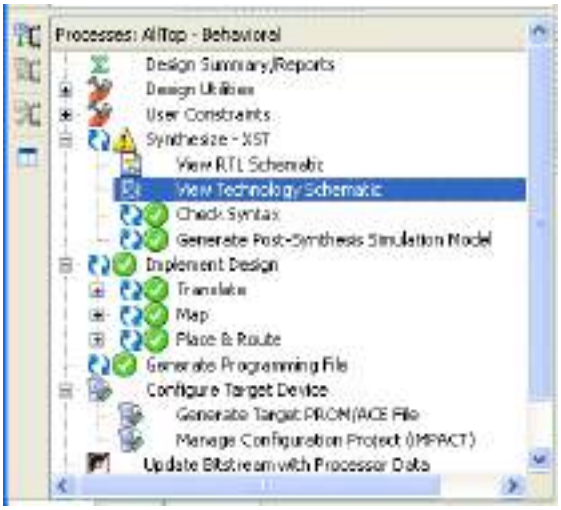


Figure 5-26: Converting a VHDL design to a hardware design processes

The green-ticks should get on all of the Synthesis, Translate, Map and Place-and-Route process items, as these processes complete. A yellow exclamation mark may get on some processes. It just means that a warning has been generated in a process.

The design has now been implemented. All that remains is the creation of the programming file for downloading, and the hardware configuration. Figure 5.27 shows the input and output of FLC as displayed on LCD screen.

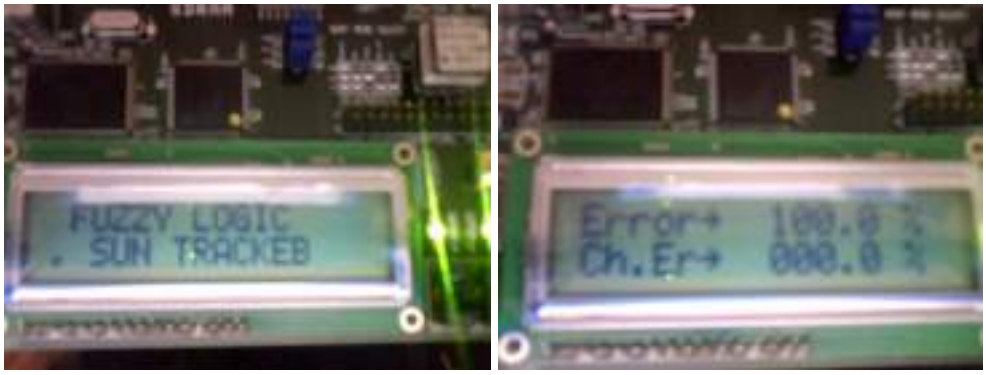


Figure 5-27: FLC I/O on LCD of the FPGA card.

Figure 5.28 shows the real complete sun tracker system .

Figure 5-28: Complete sun tracker system.

5.7 Mechanical Construction and Components:

System prototype is shown in Figure 5.29 consists of a mechanical mechanism of 2 degrees of freedom (D.O.F) designed to support and direct a PV solar cell attached to it. Mechanism has the ability to rotate the PV cell about 2_axes, x or z. But initially, we have locked z- axis rotation and applied control scheme to x- axis only.

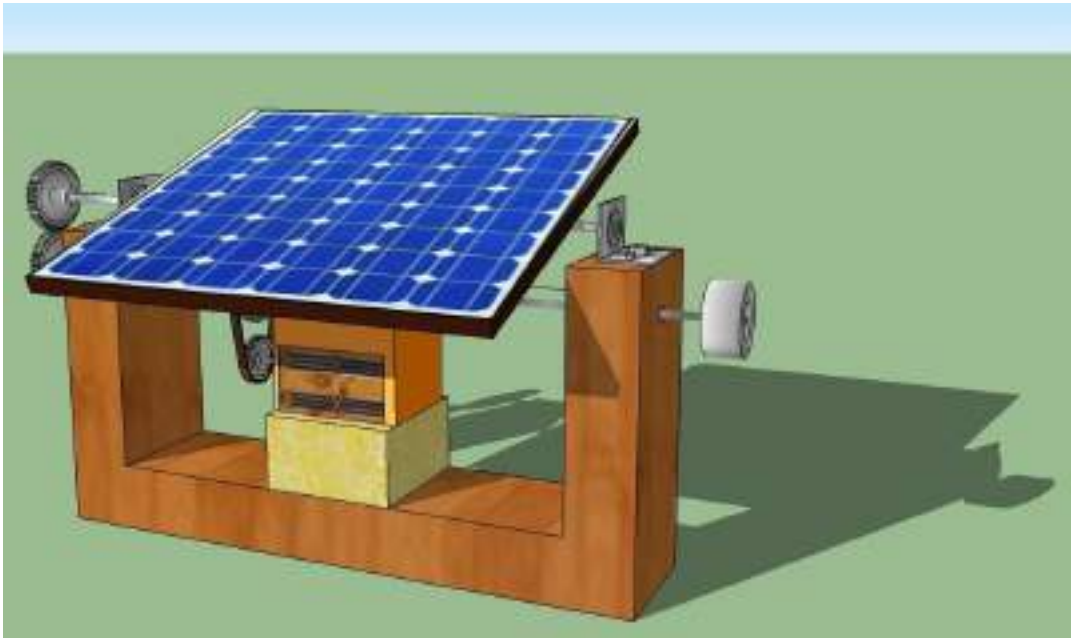


Figure 5-29 : System prototype.

Electro- mechanical drive system of x- axis consists of a stepper motor with a 1.5 cm radius pulley attached to its shaft and is driving a 2.5 cm radius pulley attached to main driving shaft as shown in Figure 5.30, through a belt. Belt mechanism realizes a speed reduction of 40% ((1-1.5/2.5)*100), and a torque increase of 40% in order to withstand demand load.

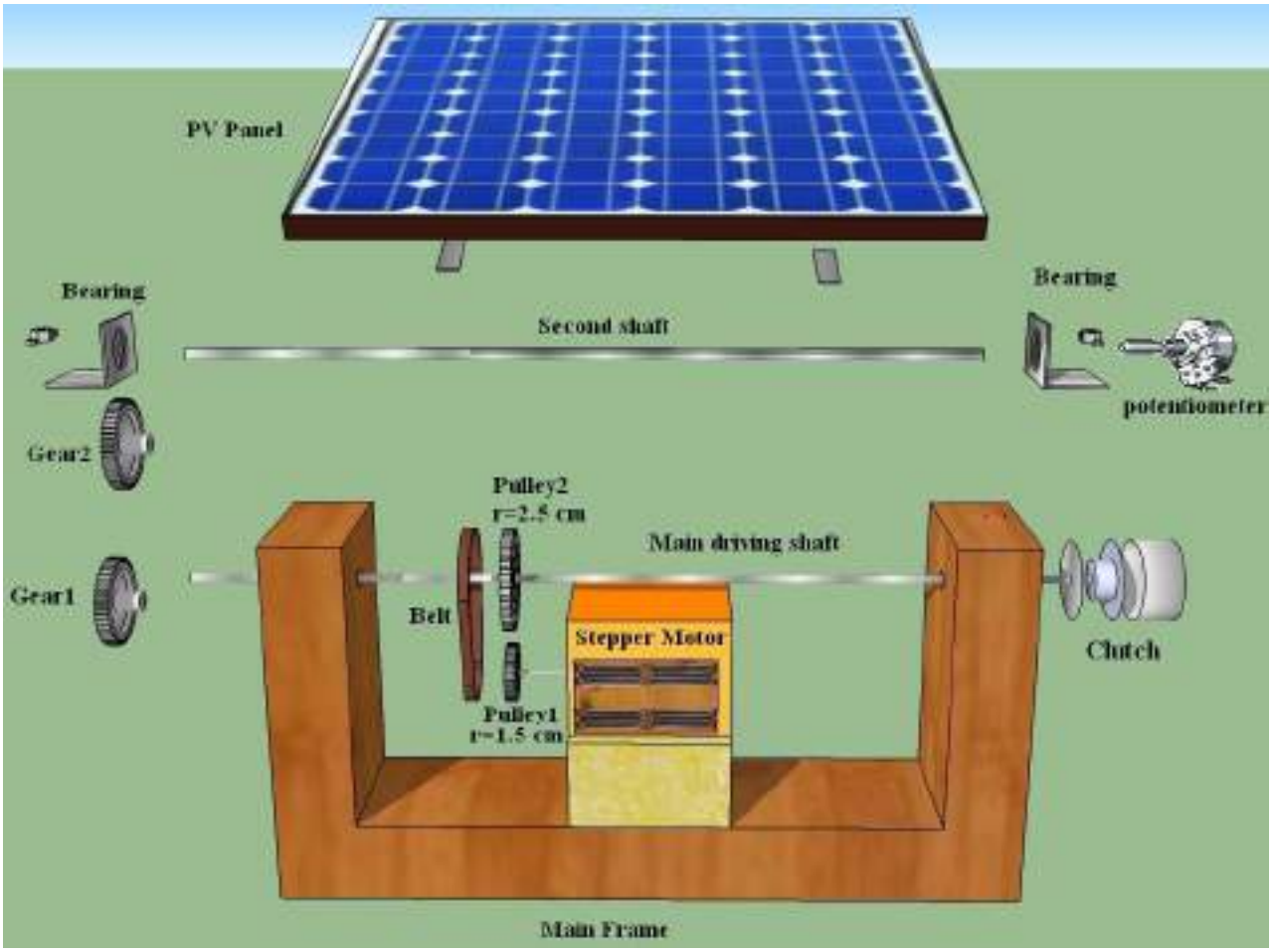


Figure 5-30: Mechanical construction and components.

Main driving shaft, shown in Figure 5.30 is attached to the main frame and supported with two bearings. Also, this shaft is provided by an electro-mechanical clutch in order to prevent axis rotation when driving motor is disabled and to assure to keep the PV panel at the same end position. Main driving shaft transmits rotation to the second shaft, shown in Figure 5.30, through two identical meshing gears with the same angular speed. Second shaft is supported by the main frame by two ball bearings. PV panel is attached to the second shaft and its angular position is measured with a potentiometer attached to the second shaft end. The system is represented graphically in real dimensions using AutoCAD program in other Figures putted in Appendix C.

5.8 Experimental Results:

The experimental data of the solar generating power system are measured outdoors in interval from 13 to 17 June, by measuring the voltage and current for the same load in each hour and calculating the average value for all days. Table 5-3 shows these values.

Table 05-3: **Experimental results of fixed angle type and tracking system**

Time of day	PV output values at fixed angle			PV output values at variable angles		
	Voltage (V)	Current(A)	Power(w)	Voltage (V)	Current(A)	Power(w)
7.00 A.M	9.9	0.06	0.594	15.2	0.22	3.344
8.00 A.M	13.4	0.253	3.3902	18.6	0.53	9.858
9.00 A.M	15.8	0.6	9.48	19.76	0.96	18.9696
10.00 A.M	19.5	0.72	14.04	20.1	0.875	17.5875
11.00 A.M	18.92	0.967	18.2956	19.3	1	19.3
11.30 A.M	18.9	0.98	18.522	19.2	1	19.2
12.00 noon	19.55	0.966	18.8853	19.6	0.98	19.208
1:00 P.M	19.62	0.91	17.8542	19.62	0.913	17.9131
2:00 P.M	19.6	0.893	17.5028	18.95	0.976	18.4952
3:00 P.M	19.71	0.873	17.2068	20.1	0.9	18.09
4:00 P.M	19.5	0.53	10.335	19.7	0.635	12.5095
5:00 P.M	19.4	0.38	7.372	20.2	0.56	11.312
5:30 P.M	18.8	0.32	6.016	19.84	0.52	10.3168
6:00 P.M	14.7	0.13	1.911	19.9	0.2	3.98
7:00 P.M	9	0.0014	0.0126	10.5	0.00152	0.01596
Sum=	**256.3**	**8.5834**	**161.4176**	**280.57**	**10.27052**	**200.0996**

Figure 5.31 represents the voltage data in the Table5-3 as a graphical curve using MS_EXEL program.

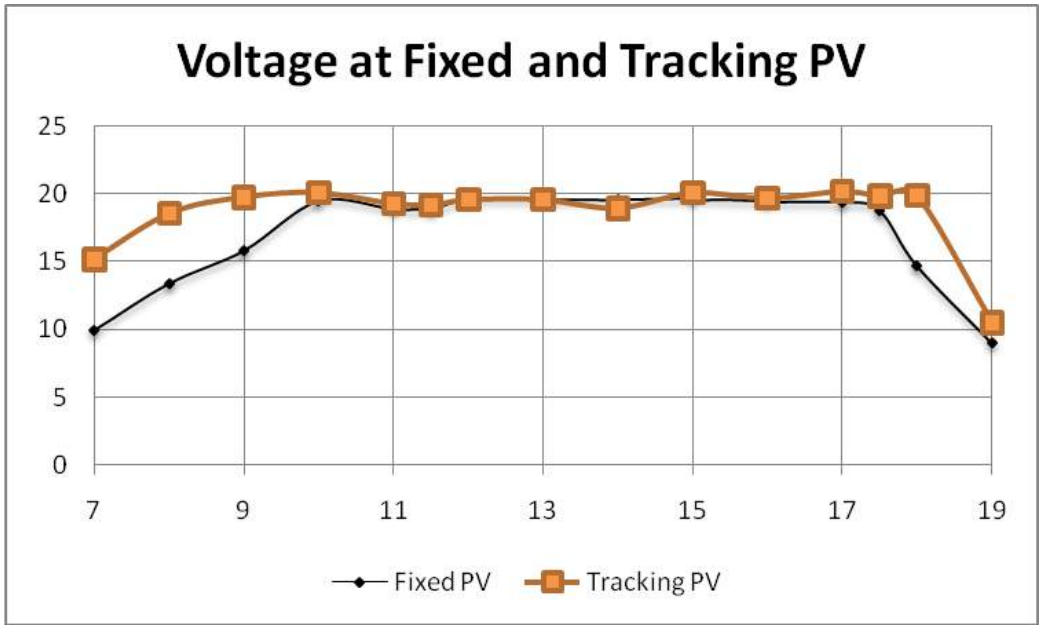


Figure 5--31: The voltage comparison of fixed angle PV and tracking PV system.

Figure 5.32 represents the current data in the Table5-3 :

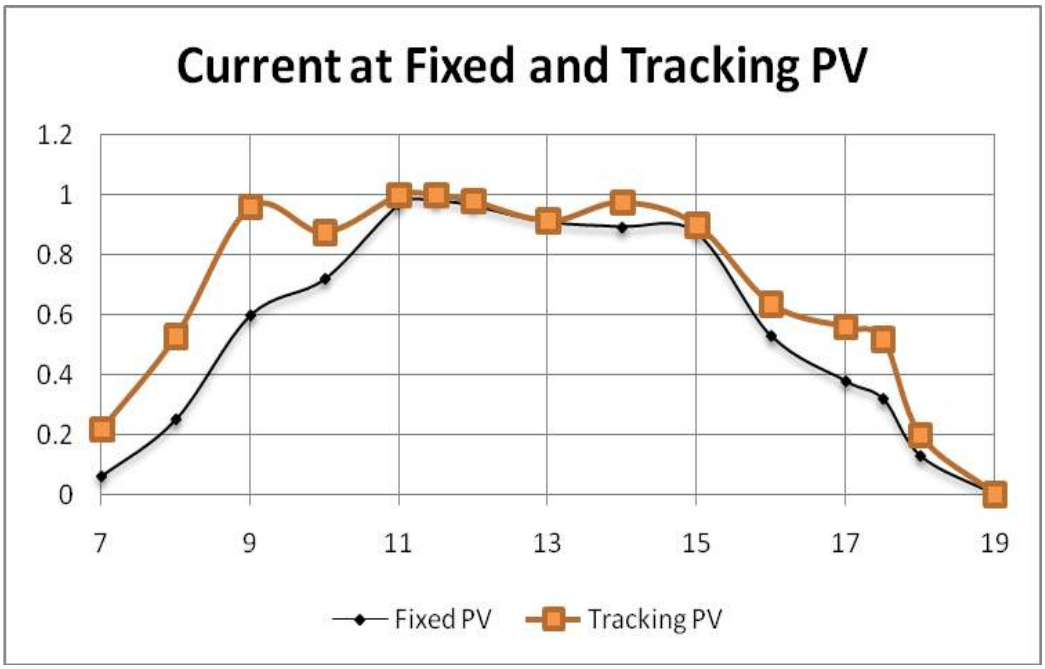


Figure 5-32: The current comparison of fixed angle PV and tracking PV system.

Figure 5.33 represents the power data in the Table5-3 :

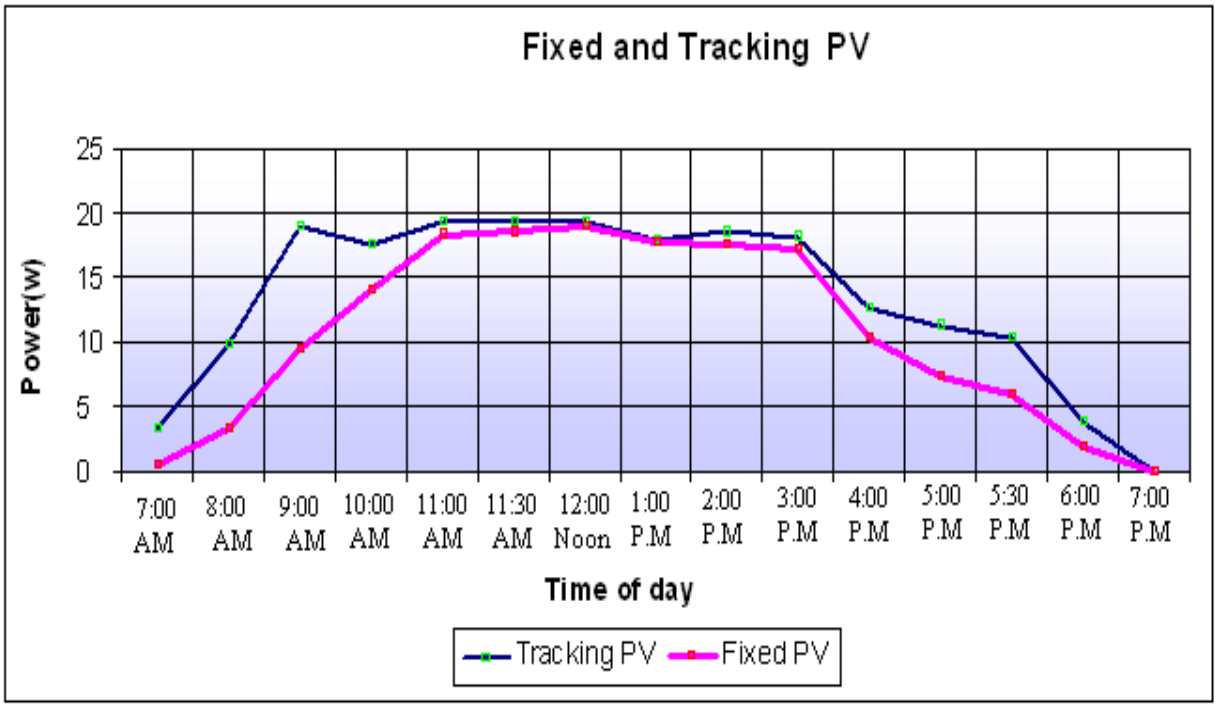


Figure 5-33: The power generation comparison of fixed angle type and tracking systems.

From Table 5.4 the efficiency can calculated as:

Efficiency= [(200.0996-161.4176)*100/161.4176]= 23.96397% ≈ 24%

that mean the efficiency with solar tracking methodology is 24% percent higher than with fixed angle. It has been shown that the sun tracking systems can collect about 24% more energy than what a fixed panel system collects and thus high efficiency is achieved through the tracker.

CHAPTER 6
Designing MPPT System Using FLC on FPGA

6.1 MPPT of PV Using Fuzzy Controller:

Maximum power point tracking system uses dc to dc converter to compensate the output voltage of the solar panel to keep the voltage at the value which maximises the output power. MPP fuzzy logic controller measures the values of the voltage and current at the output of the solar panel, then calculates the power from the relation (P=V*I) to extract the inputs of the controller. The crisp output of the controller represents the duty cycle of the pulse width modulation to switch the dc to dc converter. Figure 6.1 shows the Maximum power point tracker (MPPT) system as a block diagram.

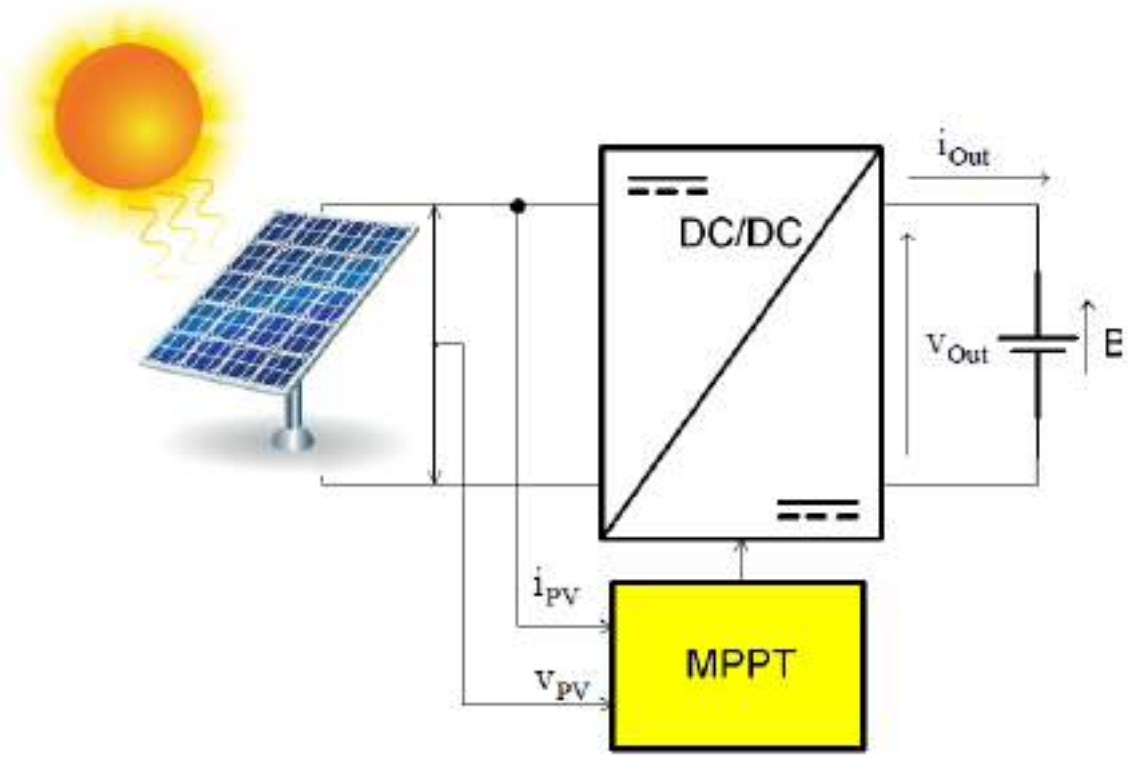


Figure 6-1 : **Maximum Power Point Tracker (MPPT) system as a block diagram.**

6.2 MPPT Fuzzy Logic Controller:

The FLC examines the output PV power at each sample (time_k), and determines the change in power relative to voltage (dp/dv). If this value is greater than zero the controller change the duty cycle of the pulse width modulation (PWM) to increase the voltage until the power is maximum or the value (dp/dv)=0, if this value less than zero the controller changes the duty cycle of the PWM to decrease the voltage until the power is maximum as shown in Figure 6.2.

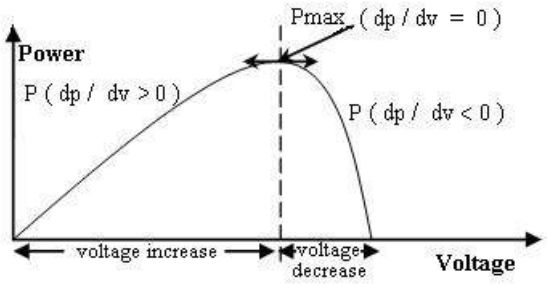


Figure 6-2: Power-voltage characteristic of a PV module.

FLC has two inputs which are: error and the change in error, and one output feeding to the pulse width modulation to control the DC-to-DC converter. The two FLC input variables error E and change of error CE at sampled times k defined by:

$$Error(k) = \frac{P(k) - P(k-1)}{V(k) - V(k-1)} \tag{6.1}$$

$$Change\_Error(k) = Error(k) - Error(k-1) \tag{6.2}$$

where P(k) is the instant power of the photovoltaic generator. The input error (k) shows if the load operation point at the instant k is located on the left or on the right of the maximum power point on the PV characteristic, while the input CE (k) expresses the moving direction of this point. The fuzzy inference is carried out by using Mamdani method, FLC for the Maximum power point tracker. FLC contains three basic parts: Fuzzification, Base rule, and Defuzzification.

- **Fuzzification**

Figure 6.3 illustrates the fuzzy set of the Error input which contains 7 Triangular memberships

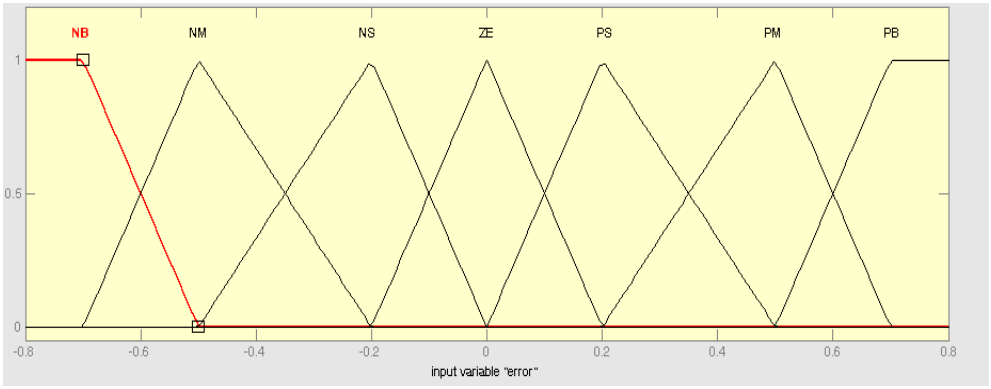


Figure 6-3: Membership function of error (E).

Figure 6.4 illustrates the fuzzy set of the Change of Error input which contains 7 Triangular memberships.

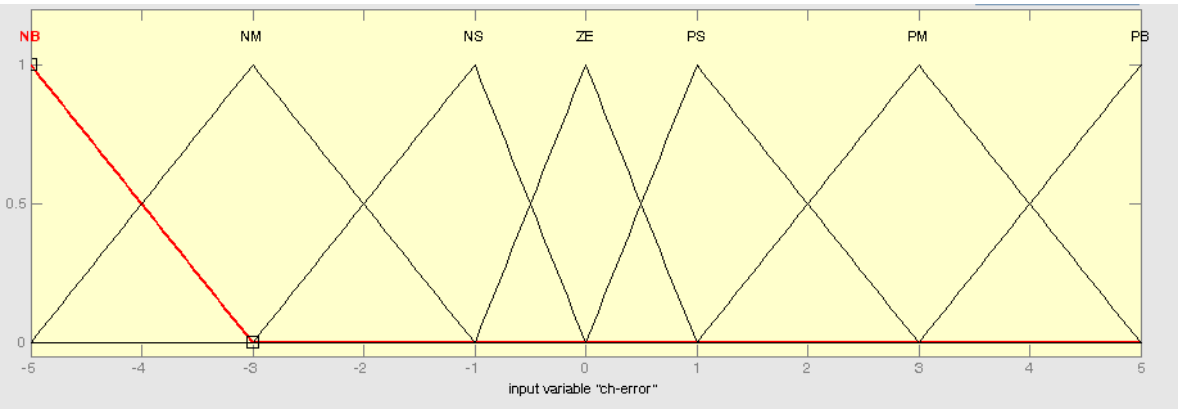


Figure 6-4:Membership function of change of error (CE).

Figure 6.5 illustrates the fuzzy set of the output which contains 7 Triangular memberships.

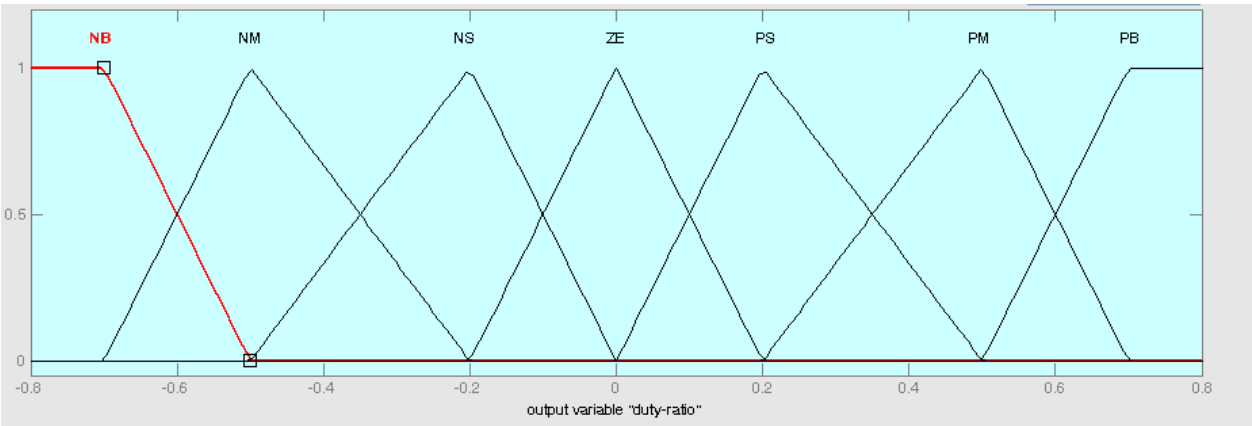


Figure 6-5: Membership function of duty ratio (D).

- **Control rule base**

The knowledge base defining the rules for the desired relationship is between the input and output variables in terms of the membership functions illustrated in Table 6.1. The control rules are evaluated by an inference mechanism, and represented as a set of:

IF Error is ... and Change of Error is ... THEN the output will … .

For example: Rule1: IF Error is NL and Change of Error is ZE THEN the output is NS.

The linguistic variables used are:

NB: Negative Big.

NM: Negative Medium.

NS: Negative Small.

ZE: Zero.

PS: Positive Small.

PM: Positive Medium.

PB: Positive Big.

Table 6-1: **Control rule base for MPPT fuzzy controller.**

$E \downarrow / CE \rightarrow$	NB	NM	NS	ZE	PS	PM	PB
NB	ZE	ZE	ZE	NB	NB	NB	NB
NM	ZE	ZE	ZE	NM	NM	NM	NM
NS	NS	ZE	ZE	NS	NS	NS	NS
ZE	NM	NS	ZE	ZE	ZE	PS	PM
PS	PM	PS	PS	PS	ZE	ZE	PS
PM	PM	PM	PM	PM	ZE	ZE	ZE
PB	PB	PB	PB	PB	ZE	ZE	ZE

Figure 6.6 shows the surface of the base rules using in FLC.

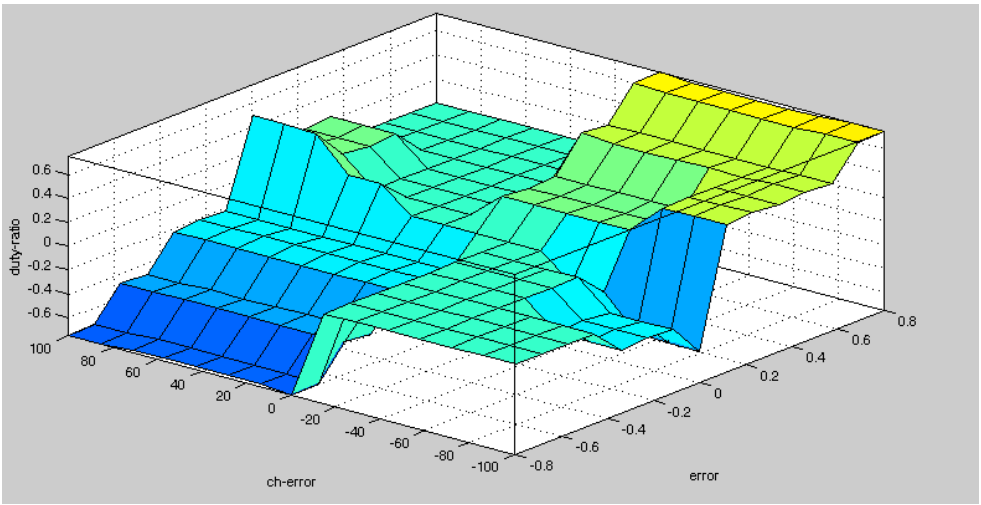


Figure 6-6: Rule surface of FLC.

- **Defuzzification**

The defuzzification uses the centre of gravity to compute the output of this FLC which is the duty cycle(D):

$$D = \frac{\sum_{j=1}^{n} \mu(d_j) - d_j}{\sum_{j=1}^{n} \mu(d_j)} \tag{6.3}$$

6.3 MPPT Fuzzy Logic Controller Simulation on Matlab/Simulink:

Before applying the fuzzy controller on PV, the modeling of PV must be set-up. As shown in Chapter2.

6.3.1 PV modeling for Simulation

The equations from 2.1 to 2.4 for generating the current by PV array are represented by MATLAB/SIMULINK as shown in Figure 6.7.

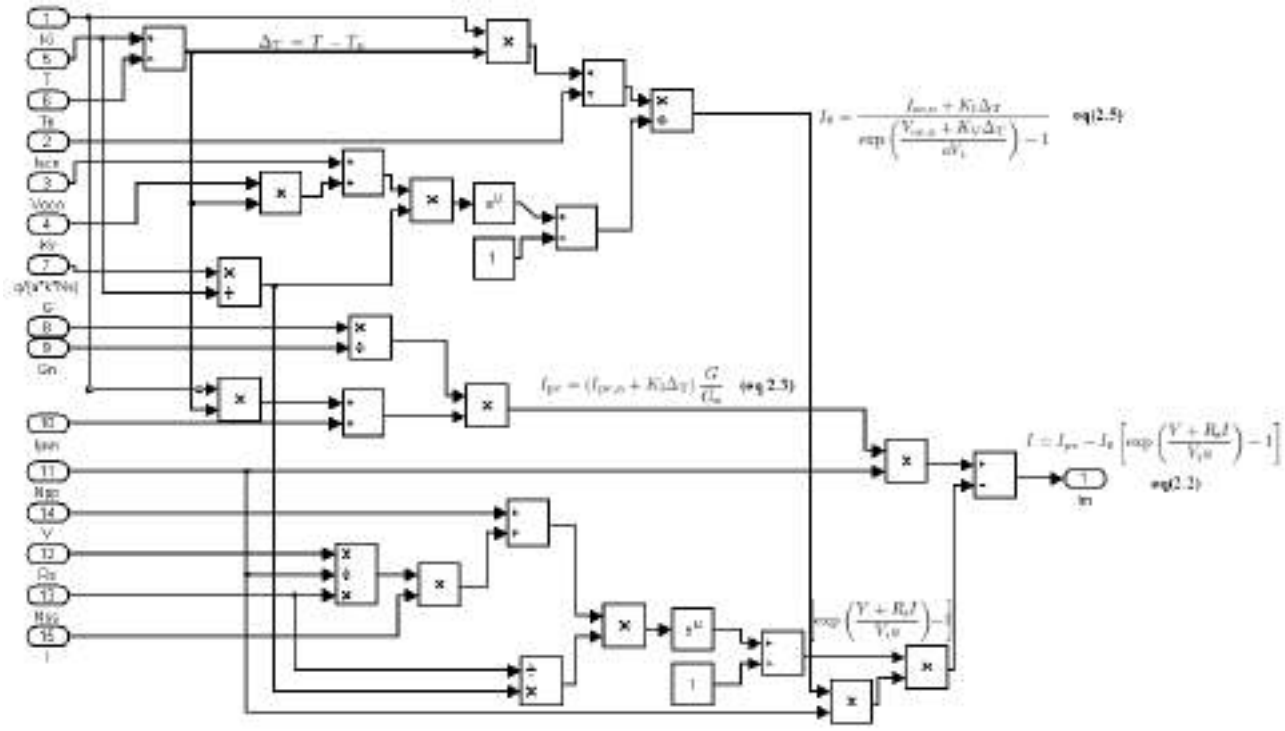


Figure 6-7: Modeling of the current generated by PV array in matlab simulink.

This current is passed through series and parallel resistors of the array as shown in Figure 2.6, then all these blocks are converted to one sub system block with two inputs (Temperature, and Irradiation) as shown in Figure 6.8.

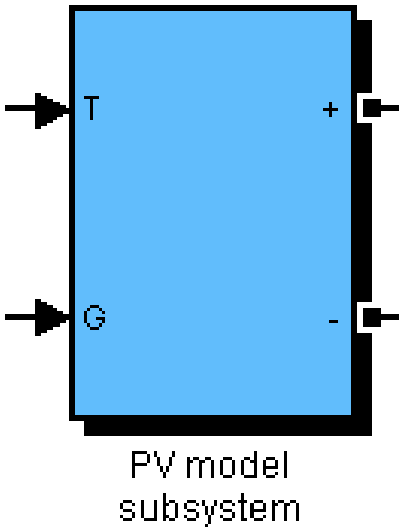


Figure 6-8: PV model Subsystem.

6.3.2 GUI Interface for PV Model:

The PV model have a large number of parameters, so a graphical user interface GUI is set-up for entering the parameters of any array model using information from its datasheet as shown in Figure 6.9.

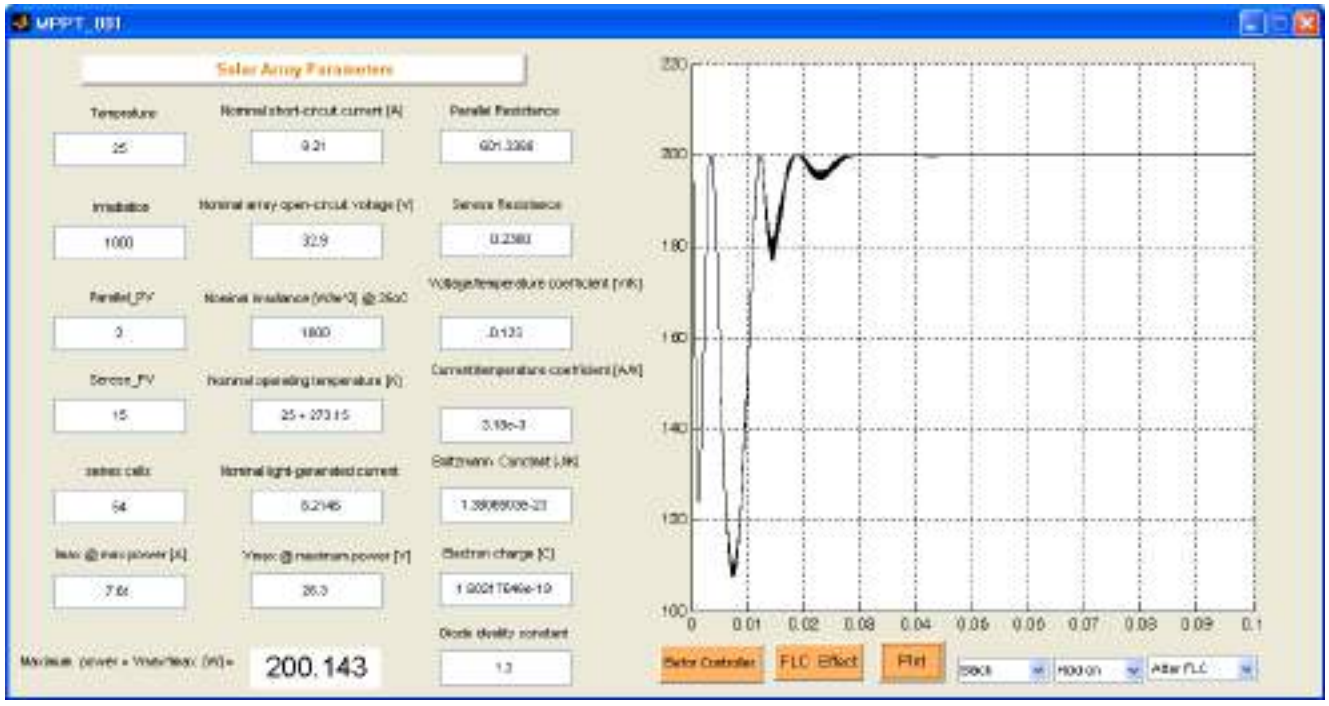


Figure 6-9: **GUI for entering the parameters of any array model from its datasheet.**

6.3.3 Control Signal Generation in Simulation

Figure 5.10 shows how the equations 6.1 and 6.2 are represented, to generate the Error and Change in error signals as inputs for the fuzzy logic controller.

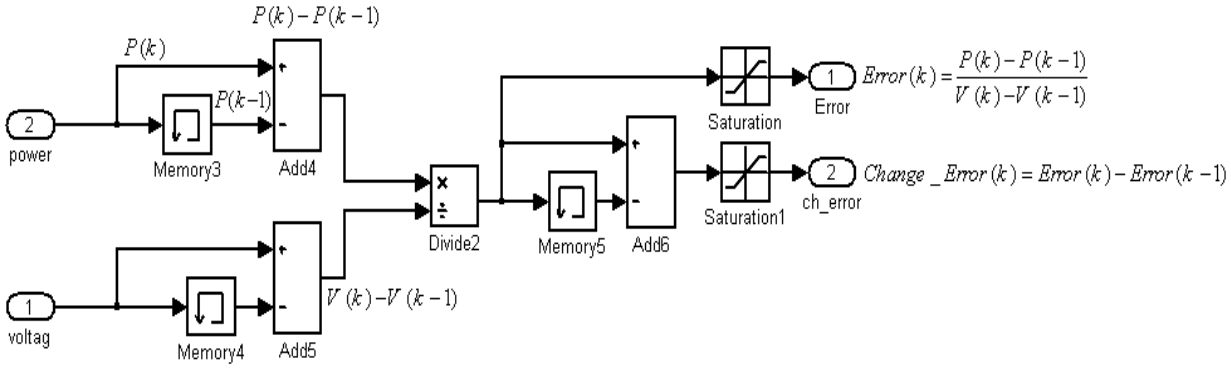


Figure 6-10: Generating the Error and Change in Error Signals.

6.3.4 Fuzzy Logic Controller Simulation:

The designed fuzzy controller now can connected between PV module and DC-to-DC converter module to tracking the MPP, as shown in Figure 6.11

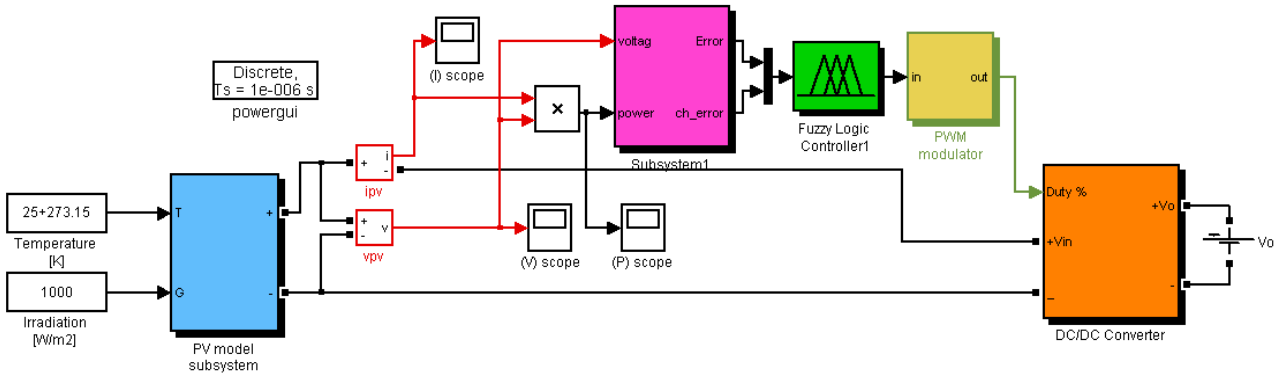


Figure 6-11: Controlling the PV power using FLC.

The parameters information of PV array is entered by GUI interface from the KC200GT solar array datasheet:

Nominal short-circuit voltage [A]: $I_{scn} = 8.21$

Nominal array open-circuit voltage [V]: $V_{ocn} = 32.9$

Array current at maximum power point [A]: $I_{mp} = 7.61$

Array voltage at maximum power point [V] : $V_{mp} = 26.3$

Voltage/temperature coefficient [V/K] : $K_v = -0.123$

Current/temperature coefficient [A/K] : $K_i = 3.18e^{-3}$

Nunber of series cells : Ns = 54

Nominal irradiance [W/m^2] at 25^{o}C : $G_n = 1000$

Nominal operating temperature [K]: $T_n = 25 + 273.15$

Boltzmann Constant [J/K]:k = $1.3806503e^{-23}$

Electron charge Constant [C]: q = $1.60217646e^{-19}$

Diode ideality constant : a = 1.3 .

Figure 6.12 shows the Characteristic P-V curve of a practical photovoltaic device with the last specifications before adding the fuzzy logic controller.

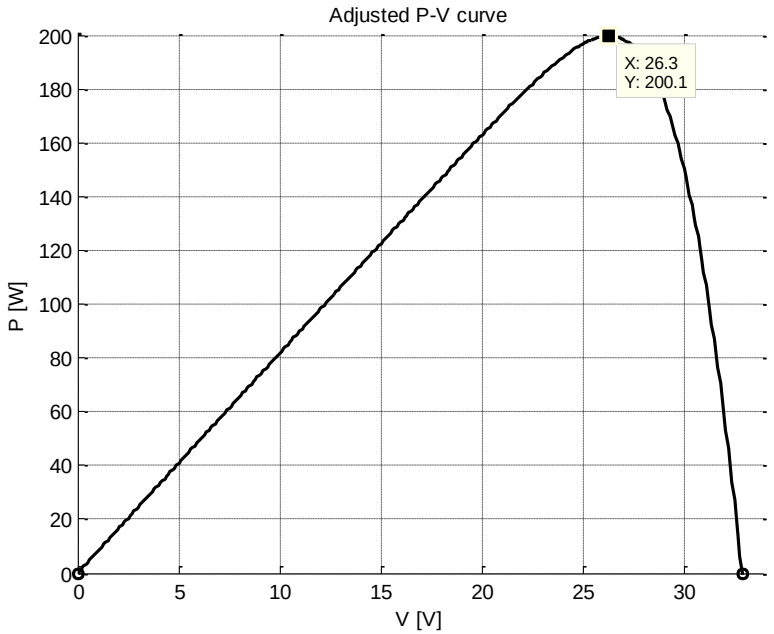


Figure 6-12: The Characteristic P-V curve before adding the FLC.

Figure 6.13 shows the effect of the FLC controller on the PV power, since it becomes constant at the maximum value (200.14 W) after a small stilling time.

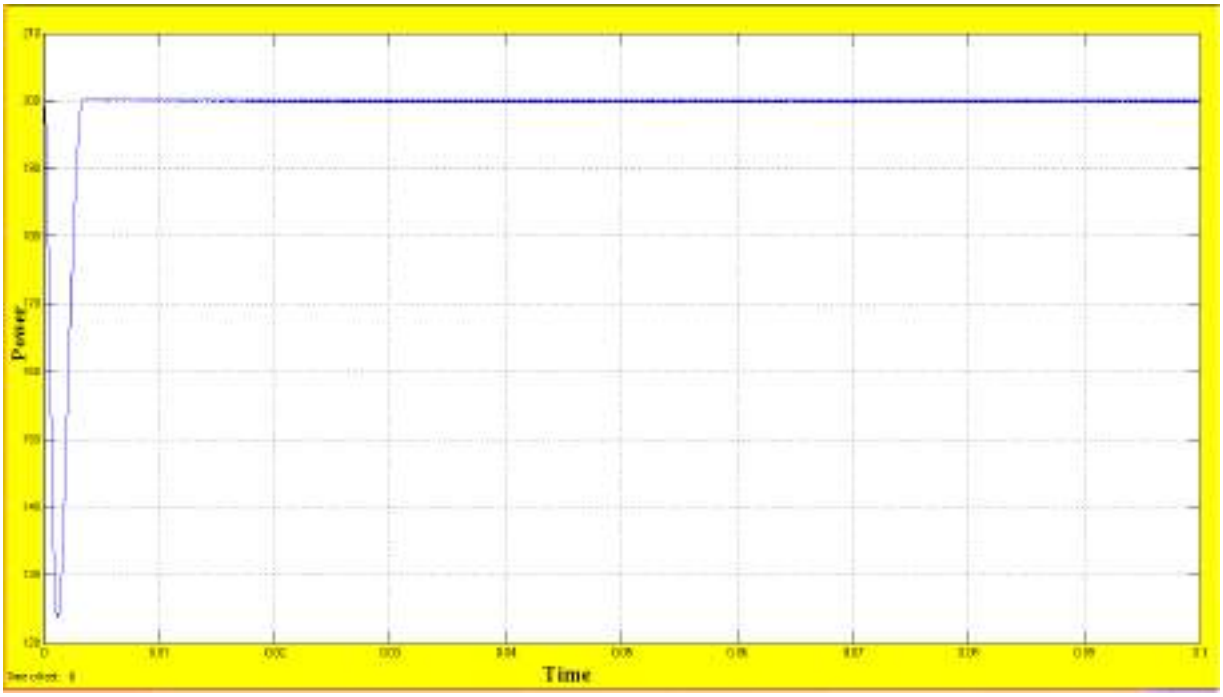


Figure 6-13: Controller effect on the power.

6.4 Comparison Between FLC and Conventional Controller:

The results of applying the FLC on PV system to track the maximum power point is compared with a conventional controller applied on the same system by Villalva [33]. This controller is perturbation and observation controller. The principle of this controller is done by changing the PWM duty cycle(D) and observing the effect on the output PV power, this can be detailed as follows:

- when $dp/dv > 0$, the voltage is increased, this is done through $D(k) = D(k-1) + C$.
(C : incrementation step),
- when $dp/dv < 0$, the voltage is decreased through $D(k) = D(k-1) - C$.

Figure 5.10 shows the effect of the two controllers perturbation and observation and FLC controller on the same PV power .

The response of FLC is better than the response of the perturbation and observation controller since it take more settling time.

Other drawback point in perturbation and observation controller is that it depend on knowing the value of the voltage at the maximum power poin (V_m).

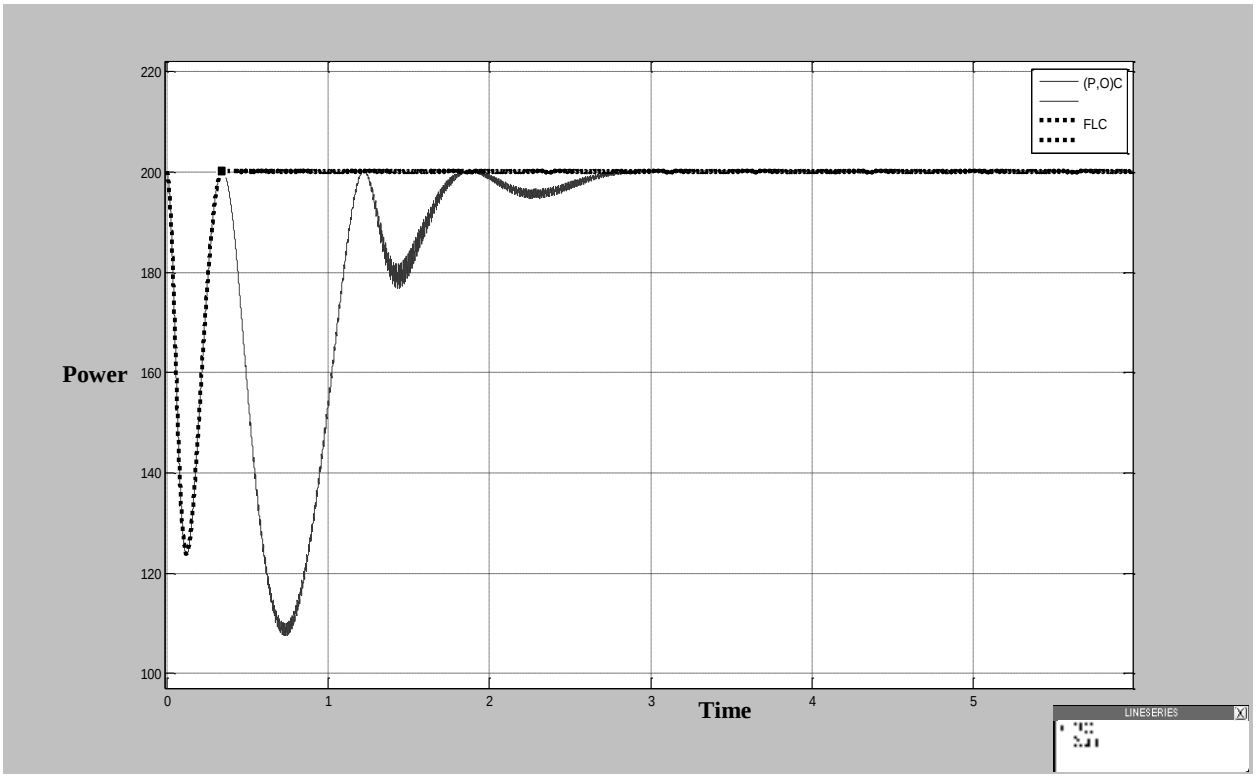


Figure 6-14: The effect of the two controllers conventional and FLC controller on the PV power .

6.5 Implementing the Maximum Power Point Tracker:

In the first, the fuzzy logic controller for MPPT will implemented on the FPGA card. Then, the DC-to-DC implemented and connected with the FPGA.

6.5.1 Implementing the FLC of MPPT on FPGA:

The FLC is implemented as the same steps in section 5. 6 on the same FPGA card. Figure 5.15 shows the RTL schematic diagram in Xilinx software RTL Viewer to view a schematic representation for the FLC and other components after implementing it on Xilinx_ISE 11.1 software. The inputs of the controller are the error and change in error as in equations 6.1 and 6.2. The output of the controller is connected with a PWM module designed on the FPGA, its looks as green block in Figure 6.15. The PWM frequency of the modulating signal is about 3KHz, this value calculated by experiment. A 14-bit counter runs at the clock of FPGA =50MHz completes cycles at a rate $50M/2^{14} \approx 3KHz$. In this case, each level in an 8-bit modulating signal corresponds to $2^{14}/256 = 2^6$ clock pulses.

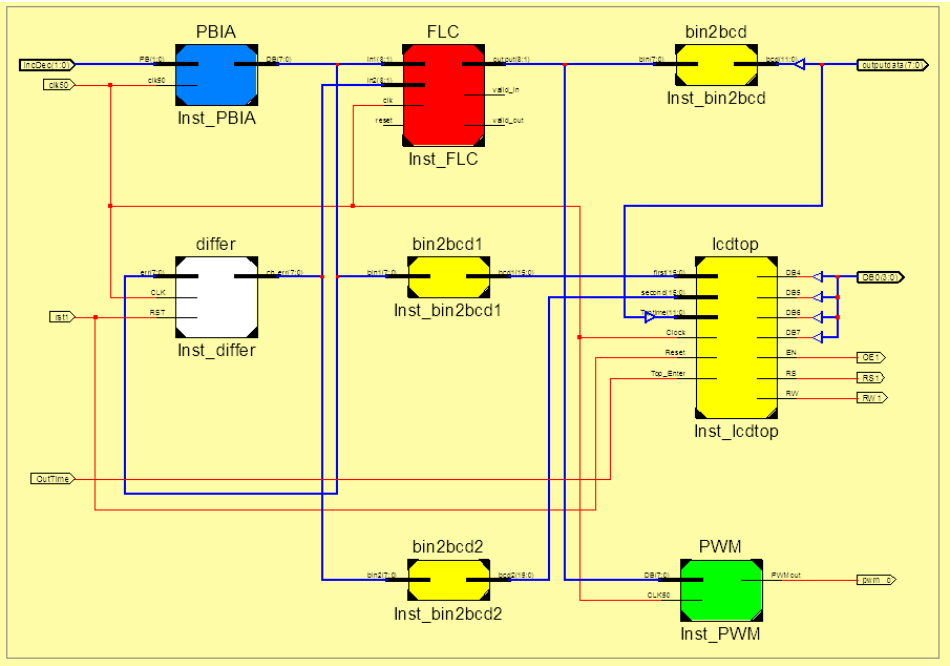
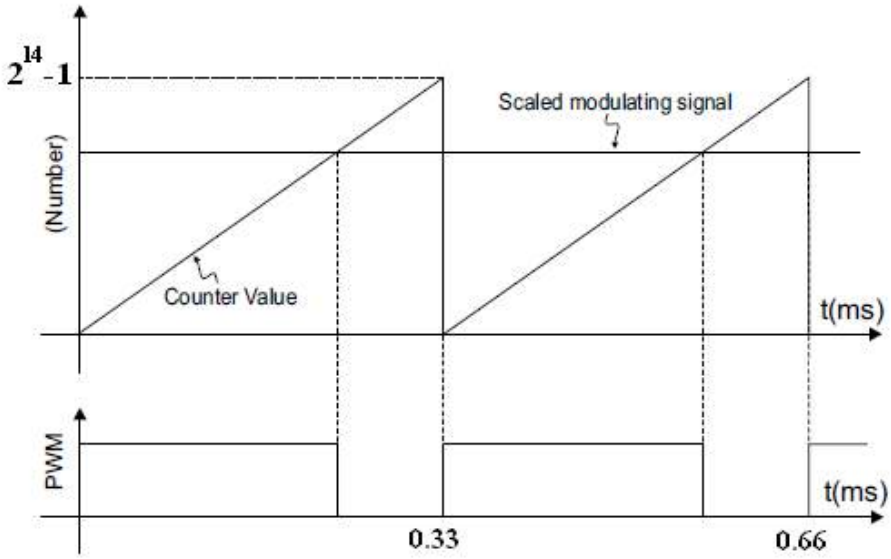


Figure 6-15: RTL schematic diagram for the FLC with other blocks.

Figure 6.16 shows how to generate the PWM signal and the VHDL code is shown in Appendix D.



.

Figure 6-16: Generating PWM signals.

The output of the PWM is examined using the oscilloscope by changing the values of the FLC and observe the change in the duty cycle of the PWM output as shown in Figure 6.17.

Figure 6-17: Examining the PWM output.

6.5.2 Implementing the DC-to-DC Converter:

The DC-to-DC converter is implemented as shown in Figure 6.18 and connected with FPGA card .

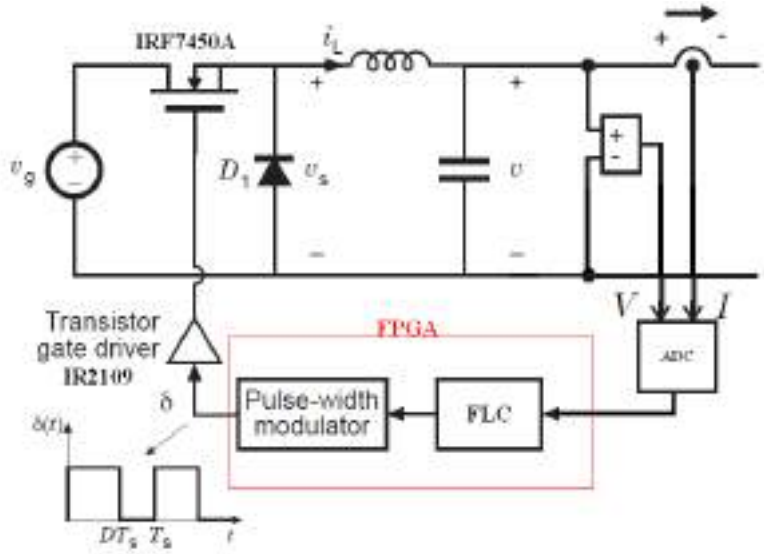


Figure 6-18: DC to DC converter.

In the hardware part, the circuit is designed to step down DC-to-DC voltage. The circuit included parts of Buck components such as controllable switch (IRF740A), inductor and capacitor, PIC16F877 microcontroller as an ADC, IR2110 Half Bridge Driver, optocubler isolator (6N137), and other basic components. In order to maintain output voltage, controller

will be operated in feedback circuit. The details about some of these components is shown in Appendices (E, F, G). When the duty cycle is in ON state, the circuit become as in Figure 6.19, diode become as reversed biased and the inductor will deliver current and switch conducts inductor current. The current through the inductor increase, as the source voltage would be greater.

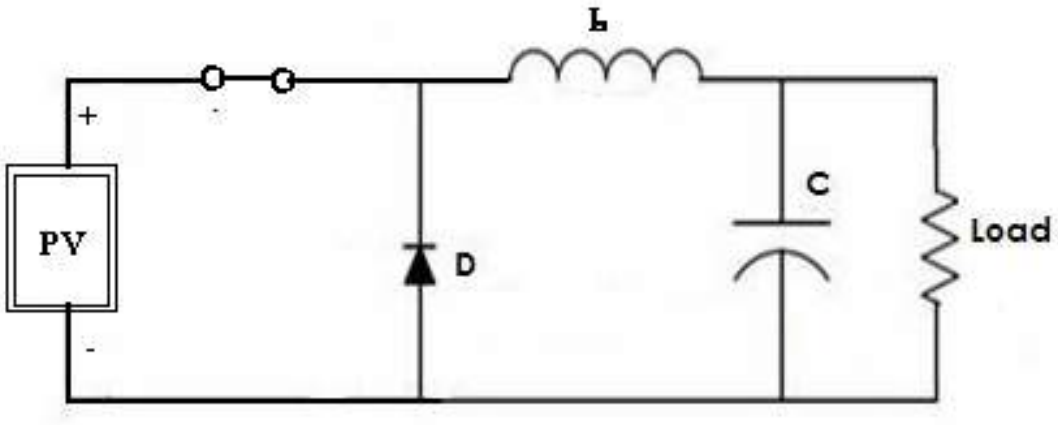


Figure 6-19: Equivalent circuit for switch closed.

The energy stored in inductor increased when the current increase, and the inductor acquires energy. Capacitor will provides smooth out of inductor current changes into a stable voltage at output voltage .

When the duty cycle is in OFF state, The circuit become as in Figure 6.20, diode is ON and the inductor will maintains current to load. Because of inductive energy storage, inductor current will continues to flow.

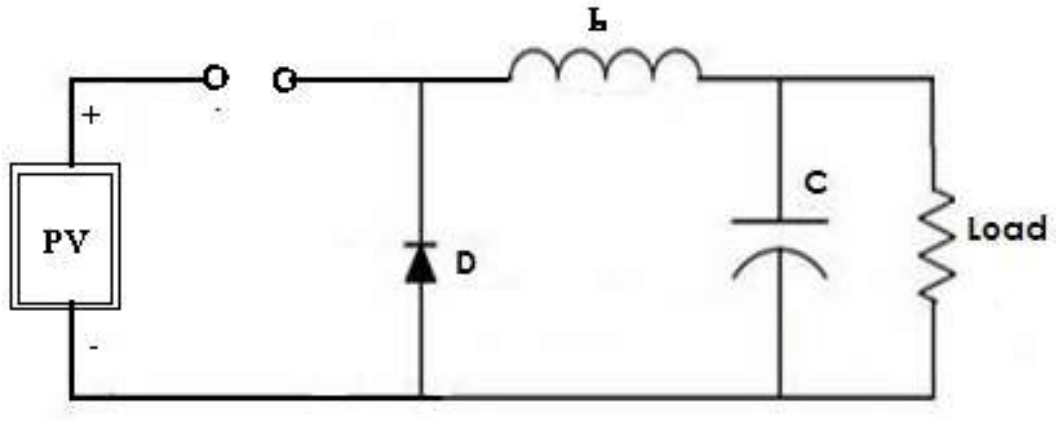


Figure 6-20: Equivalent circuit for switch open.

While inductor releases current storage, it will flows to the load and provides voltage to the circuit. The diode is forward biased. The current flow through the diode where is inductor voltage is equal with negative output voltage.

The output of the DC-to-DC converter is examined using the oscilloscope by changing the values of the FLC inputs as an open loop and observe the change in the duty cycle of the PWM output and the change in the converter output as shown in Figure 6.21.

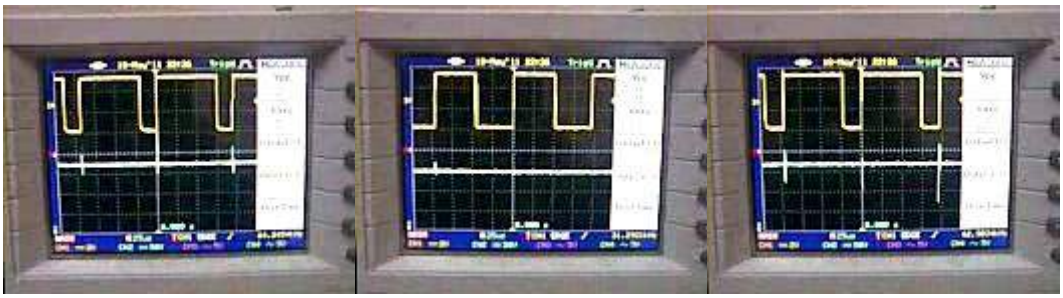


Figure 6-21: Examining the PWM output with open loop controller.

After adding the close loop FLC by changing the value of the input voltage to the DC-to-DC converter, the duty cycle value is constant for each input and the output voltage is constant for all DC-to-DC input voltages as shown in Figure 6.22.

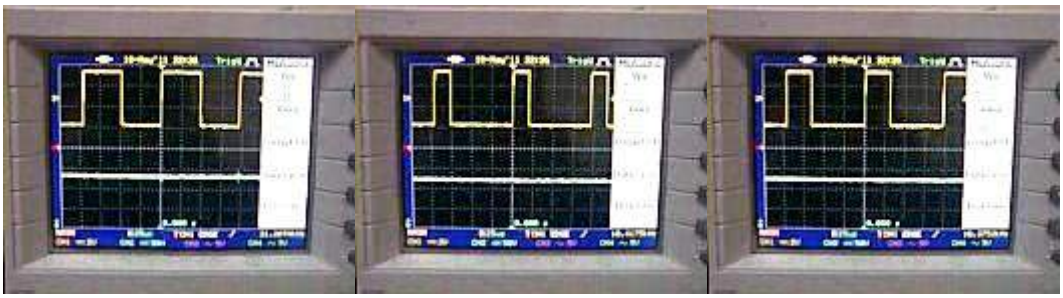


Figure 6-22: Examining the PWM output with close loop controller.

From the last results, the proposed maximum power point tracker is suitable to used to keep the PV output power at the maximum for increasing efficiency of it.

6.5.3 Experimental Results:

The experimental data of the solar generating power system are measured outdoors in interval from 13 to 17 June, by measuring the voltage and current for the same load in each hour and calculating the average value for all days. Table 6-2 shows these values.

Table 06-2: Comparison of the experimental results obtained with and without using MPPT & sun tracker (ST) controller.

Time of day	PV output values without MPPT & ST			PV output values with ST			PV output values with ST & MPPT		
	Voltage (V)	Current (A)	Power (w)	Voltage (V)	Current (A)	Power (w)	Voltage (V)	Current (A)	Power (w)
7:00 A.M	9.9	0.06	0.594	15.2	0.22	3.344	12.4	0.387	4.8
8:00 A.M	13.4	0.253	3.3902	18.6	0.53	9.858	17.5	0.63	11.1
9:00 A.M	15.8	0.6	9.48	19.76	0.96	18.969	17.8	1.067	19
10:0 A.M	19.5	0.72	14.04	20.1	0.875	17.587	18.4	1.027	18.9
11:0 A.M	18.92	0.967	18.2956	19.3	1	19.3	18.1	1.077	19.5
11:30 A.M	18.9	0.98	18.522	19.2	1	19.2	18.1	1.077	19.5
12.0 non	19.55	0.966	18.8853	19.6	0.98	19.208	18.2	1.0714	19.5
1:00 P.M	19.62	0.91	17.8542	19.62	0.913	17.913	18.1	1.0607	19.2
2:00 P.M	19.6	0.893	17.5028	18.95	0.976	18.495	17.9	1.078	19.3
3:00 P.M	19.71	0.873	17.2068	20.1	0.9	18.09	18	1.06	19.1
4:00 P.M	19.5	0.53	10.335	19.7	0.635	12.509	18.3	0.748	13.7
5:00 P.M	19.4	0.38	7.372	20.2	0.56	11.312	18.5	0.66	12.3
5:30 P.M	18.8	0.32	6.016	19.84	0.52	10.316	18.5	0.65	12.025
6:00 P.M	14.7	0.13	1.911	19.9	0.2	3.98	18	0.4	7.2
7:00 P.M	9	0.0014	0.0126	10.5	0.0015	0.0159	9	0.0057	0.052
Sum=	**256.3**	**8.5834**	**161.417**	**280.5**	**10.270**	**200.1**	**256.8**	**12.011**	**215.17**

Figure 6.23 represents the voltage data in the Table 6-2 as a graphical curve using MS_EXEL program.

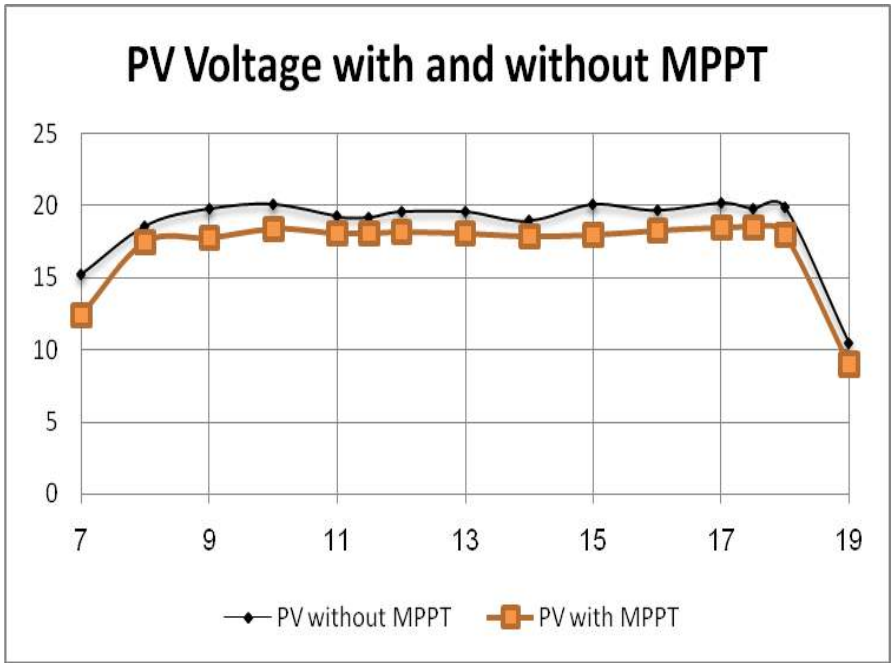


Figure 6--23: The voltage comparison with and without using MPPT controller.

Figure 6.24 represents the current data in the Table 6-2:

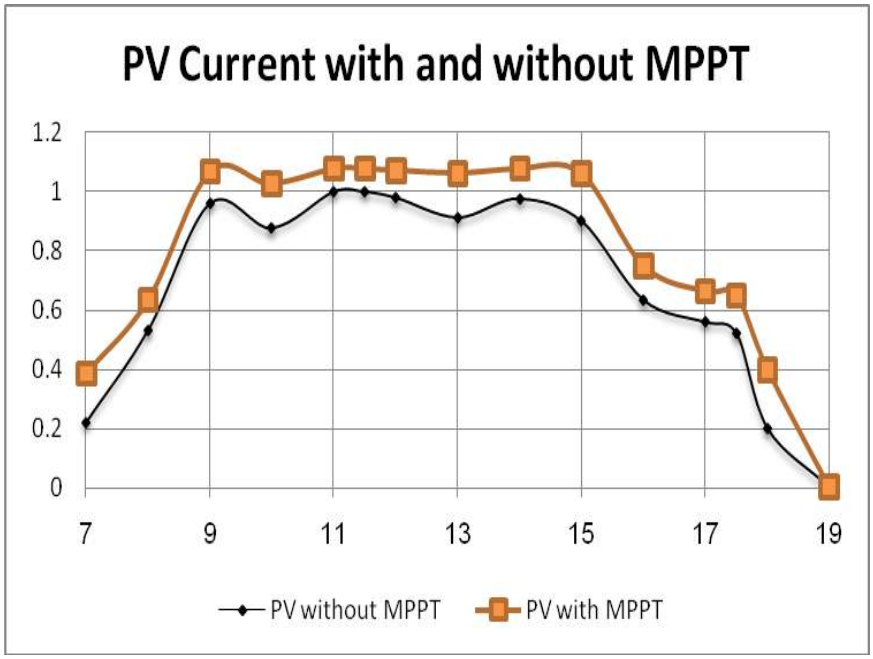


Figure 6-24: The current comparison with and without using MPPT controller.

Figure 6.25 represents the power data in the Table 6-2:

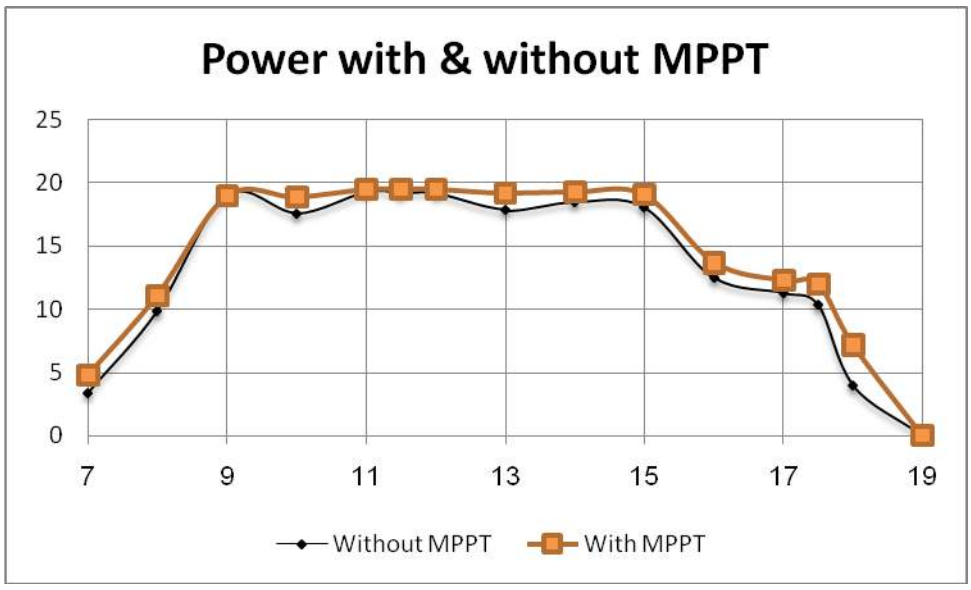


Figure 6-25: The power generation comparison with and without using MPPT controller.

From Table 6-2 the efficiency can calculated as:

$$\text{Efficiency} = [(215.177 - 200.0996) * 100 / 200.0996] \approx 7.5\%$$

that mean the efficiency with MPPT methodology is 7.5% percent higher than without MPPT. Figure 6.26 shows the comparison between the PV power before adding any controller and with sun tracker controller and with MPPT controller:

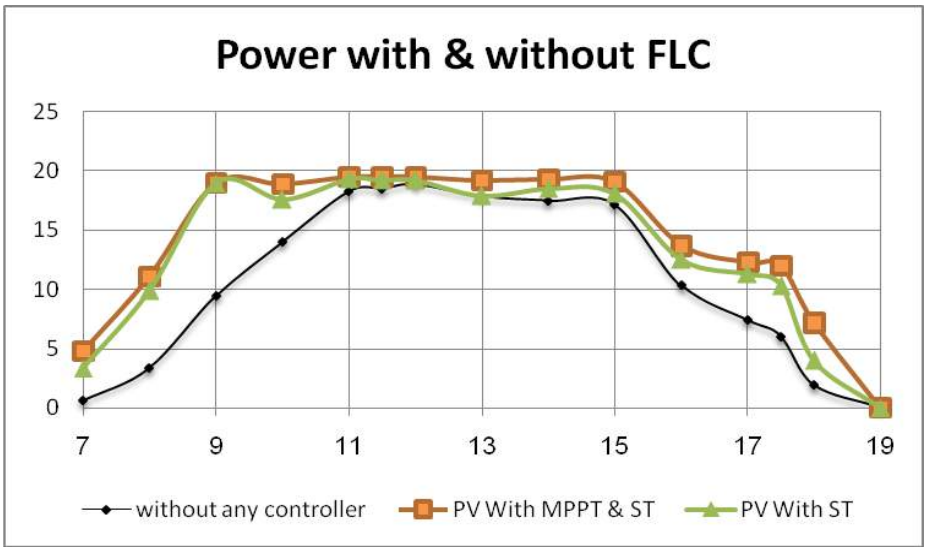


Figure 6-26: The power generation comparison with and without using ST & MPPT controller.

The efficiency with two controllers MPPT and sun tracker can calculated as:

$$\text{Efficiency} = [(215.177-161.4176)*100/161.4176] \approx 33.3\%$$

It has been shown that the sun tracking systems can collect about 33.3% more energy than what a without controllers and thus high efficiency is achieved through two trackers.

CHAPTER 7

Conclusion and Scope for Future Work

7.1 Conclusion

In Gaza Strip we have a big problem in electrical power generation, since our sources don't cover all people requirements, and the renewable energy sources such as solar energy play an important role in electric power generation, it is clean and unlimited.

In this thesis, fuzzy logic controller FLC was designed to maximizing the energy received from solar cells by two methods.

The first method is by implementing a sun tracker controlled by fuzzy logic controller to keep the PV panel pointing toward the sun by using a stepper motor. The use of stepper motor enables accurate tracking of the sun, LDR resistors are used to determine the solar light intensity. This controller has been tested using Matlab/Simulink program. The proposed solar tracking power generation system was able track the sun light automatically, so it is an efficient system for solar energy collection. It was shown that the sun tracking systems is 24% more energy efficient than a fixed panel system.

The other proposed method is by implementing a maximum power point tracker controlled by fuzzy logic controller and using buck DC-to-DC converter to keep the PV output power at the maximum point all the time. This controller was tested using Matlab/Simulink program, and the results was compared with a perturbation and observation controller applied on the same system. The comparison show that the fuzzy logic controller was better in response and don't depend on knowing any parameter of PV panel.

Electronic circuits used to implement this system were designed with minimal number of components and were integrated onto a printed circuit board for simple assembly.

After examining the solar system, it can be said that the proposed sun tracking solar array system and MPPT is a feasible methods of maximizing the energy received from solar cells.

In this thesis, the fuzzy logic control demonstrates good performance. Furthermore, fuzzy logic offers the advantage of faster design, and emulation of human control strategies. Also fuzzy control worked well for nonlinear system and shown higher efficiency over the covenantal controllers. The fuzzy logic control required complete knowledge of the operation of the system by the designer.

7.2 Scope for Future Work

In this thesis, Mamdani method was used to implement the fuzzy control rules; it can be replaces by Sugeno approach and compare it with Mamdani. Also a good area of research is using optimization method to reduce the rules of the controller such as using Genetic Algorithm with fuzzy controllers. They can be used in the control algorithm to tune the membership functions so that the inexact reasoning characteristics of the FLC are sufficient to control a system that requires precise control actions.

References:

[1] Singh,B., Goyal,R., Kumar,R. & Singh,R. (December 2009). Design and VLSI implementation of Fuzzy Logic Controller. *(IJCNS) International Journal of Computer and Network Security*, Vol. 1, No. 3,

[2] Runghim,T., Intajag,T. & Krongratana,S. Fuzzy logic PID controller based on FPGA for process control. *IEEE,* Vol.2, No.11, 2004, 1495- 1500.

[3] Gene,S. & Monroe,J. (2006). *Robust Fuzzy Controllers Using FPGAs* . NASA LaRC.

[4] Daniel,A., Pritchard,D. (Aug1983). Sun Tracking by Peak Power Positioning for Photovoltaic Concentrator Arrays, *IEEE Transactions on Control System,* Volume 3, Issue3, Page(s):2-8.

[5] Saxena,A. & Dutta,V. (July 1991). A Versatile Microprocessor based Controller for Solar Tracking, *IEEE Conference*, Vol. 2, 21 – 25May, 1990, Page(s):1105-1109.

[6] Konar,A. & Mandal,A. Microprocessor based Sun Tracker, *IEEE Proceedings*-A, Vol. 138, No.4, Page(s):237-241.

[7] Zeroual,A. Raoufi,M. Ankrim,M. & Wilkinson,A.J. (1998). Design and construction of a closed loop Sun Tracker with Microprocessor Management, *International Journal on Solar Energy*, Vol. 19, Page(s): 263- 274.

[8] Huang,F. Tien,D. and Or,J. (Dec. 1998). A microcontroller based automatic sun tracker combined with a new solar energy conversion unit, *IEEE Proceedings on Power Electronic Drives and Energy Systems for Industrial Growth,* Volume 1, 1-3, Page(s):488 - 492 .

[9] Koutroulis,D, Kalaitzakis,E. Voulgaris,K. & C,N. (Jan. 2001). Development of a microcontroller-based, photovoltaic maximum power point tracking control system, *IEEE Transactions on Power Electronics,* Volume 16, Issue 1, Page(s):46 – 54

[10] Yousef,H. (July 1999) Design and implementation of a fuzzy logic computer-controlled sun tracking system, *Proceedings of the IEEE International Symposium on Industrial Electronics,* Volume 3, 12-16, Page(s):1030 – 1034.

[11] Chong,C. Barker,S. & Chang,K. (Jan.2000). Architectures and Algorithms for Track Association and Fusion, *IEEE Transaction,* Volume15, Issue 1, , Page(s):5-13.

[12] Kuo,Y. Liang,T. & Chen,J. (June 2001). Novel Maximum-Power-Point-Tracking Controller for Photovoltaic Energy Conversion System. *IEEE Transactions on Industrial Electronics,* Volume 48, Issue 3, Page(s):594 – 601.

[13] Tse,K. Ho,M. Chung,H. & Hui,S. (Nov. 2002). A novel Maximum Power Point Tracker for PV Panels using Switching Frequency Modulation, *IEEE Transactions on Power Electronics,* Volume 17, Issue 6, Page(s):980–989.

[14] Chung,H. Tse,K. Hui,S. Mok,C. & Ho,M. (May 2003) A novel maximum power point tracking technique for solar panels using a SEPIC or Cuk converter, *IEEE Transactions on Power Electronics,* Volume 18, Issue 3, Page(s):717 – 724.

[15] Kobayashi,K. Matsuo,H. & Sekine,Y. (June 2004). A novel optimum operating point tracker of the solar cell power supply system, *IEEE Conference on Power Electronics,* Volume 3, 20-25,Page(s):2147 - 2151 Vol.3.

[16] Piao,G. Park,M. Kim,H. Cho,B. & Baek,L. (Sept 2005) A Study on the Tracking Photovoltaic System by Program Type[C], *Electrical Machines and Systems,Proceedings of the Eighth International Conference on* Vol 2, Pages: 971 – 973.

[17] Khalil,A. & El-Singaby,M. (*2003*). Position control of sun tracking system Circuits and Systems, *Proceedings of the 46th IEEE International Midwest Symposium*,Vol. 3, December 27-30, , pp. 1134 – 1137.

[18] Armstrong,S. & Hurley,W. (2005) Investigating the Effectiveness of Maximum Power Point Tracking for a Solar System, *IEEE Conference on Power Electronics,* Page(s):204 – 209.

[19] Wai,R. Wang,W. & Jun-You,M. (Nov. 2006). Grid- Connected Photovoltaic Generation System with Adaptive Step-Perturbation Method and Active Sun Tracking Scheme, *IEEE Transactions on Industrial Electronics,* Page(s):224 – 228

[20] Sungur,C. (2007) Sun –Tracking System with PLC Control for Photo-Voltaic Panels, *International Journal of Green Energy*, Vol.4, Page(s): 635–643.

[21] Ze,C. Hongzhi,Y. Ying,S. (2010). FPGA-based PV systems fuzzy MPPT control algorithm, *Fuzzy Systems and Knowledge Discovery (FSKD), Seventh International Conference,* page(s): 1244 – 1248

[22] Messai,A. Mellit,A. (October 2009). FPGA-based implementation of a fuzzy controller (MPPT) for photovoltaic module, *international conference on Energy & environment.*

[23] Borenstein,S. (January 2008) The Market Value and Cost of Solar Photovoltaic Electricity Production, *Berkeley, California* 94720-5180.

[24] El-Ashry,M. (2010). *Renewables 2010 Global Status Report*, (Paris: REN21 Secretariat). Copyright Deutsche (GTZ) GmbH.

[25] Janssen,R. (October 2002). "*Renewable Energy..into the Mainstream*". Copyright Novem SITTARD The Netherlands.

[26] Galloway,T. (2004). *Solar House: A Guide for the Solar Designer.* Architectural Press is an imprint of Elsevier, First published.

[27] WORLD ENERGY ASSESSMENT. (September 2000). *ENERGY AND THE CHALLENGE OF SUSTAINABILITY* , New York, NY 10017, First printing.

[28] Clean Energy Decision Support Centre. (2001 - 2004) *PHOTOVOLTAIC PROJECT ANALYSIS.* Minister of Natural, Resources Canada.

[29] The Solar Panel How To Guide. (December, 2010). Retrieved from http://www.solarpanelsbook.com

[30] Quaschning,V. (2005). *Understanding Renewable Energy Systems*. Volker Quaschning , London, Carl Hanser Verlag GmbH & Co KG,.

[31] Parallax. (2004). *Experiments with Renewable Energy Student Guide*. version1.

[32] Patel,R. (1999), *Wind and Solar Power Systems*. CRC Press LLC.

[33] Villalva,M. Gazoli,J. and Ruppert ,E. (MAY 2009). MODELING AND CIRCUIT-BASED SIMULATION OF PHOTOVOLTAIC ARRAYS, *IEEE TRANSACTIONS ON POWER ELECTRONICS*, VOL. 24, NO 5.

[34] Aït,M. Cheikh,S. C.Larbes, (Septembre 2007) *Maximum power point tracking using a fuzzy logic control scheme*.

[35] Liu,D. Abonyi,J. (2005). *SOFTWARE FOR FUZZY LOGIC CONTROL*, New York: Oxford.

[36] Zadeh,L. A. (1965). Fuzzy sets," Information and control, vol. 8, pp. 338-353.

[37] Zadeh,L. A. (1984). Making computers think like people, *IEEE Spectrum*, vol. 21, no. 8, pp. 26-32,.

[38] Zadeh,L. A. (1984). Fuzzy logic = Computing with words, *IEEE Trans. Fuzzy Systems,* vol. 4, pp. 103-111, 1996.

[39] Wang,L.X. (1997). *A Course in Fuzzy Systems and Controls*. Englewood Cliffs: Prentice-Hall,.

[40] E H Mamdani, "Application of Fuzzy algorithms for the control of a dynamic plant" *IEEE* vol. 121, pp. 1585-1588, 1974.

41] Bonissone,P. (1997, July 10). Fuzzy Logic and Soft Computing: Technology Development and Applications, *General Electric CRD, Schenectady* NY 12309, USA.

[42] Jain,L. Martin,N. (1999). *Fusion of Neural Networks, Fuzzy Sets, and Genetic Algorithms*. Industrial Applications, Boca Raton: CRC PRESS.

[43] Mobaied,M. (2008). *Fuzzy Logic Speed Controllers Using FPGA Technique for Three-Phase Induction Motor Drives*. Master of Science in Electrical Engineering.

[44] Riza,B. Sheldon,T. (1997). Fuzzy Systems Design Principles Building Fuzzy IF-THEN Rule Bases, *IEEE* PRESS.

[45] Funk,P. (2005). *Online fuzzy case-based individual stress diagnosing system*, Master's Thesis of Science in Technology, MÄLARDALEN University.

[46] Jamshidi,M. Vadiee,N. and Ross,T. (1993). Fuzzy Logic and Control Software and Hardware Applications, *Englewood Cliffs: Prentice Hall, Inc*.

[47] Kaehler,S. (2009) "Fuzzy Logic Tutorial ,Published at Encoder, the newsletter of the Seatte Robotics Society", URL:http://www.seattlerobotics.org/encoder/mar98/fuz/flindex.html

[48] Passino,K. Yurkovich,S. (1998). *Fuzzy Control*, Longman: Addison Wesley.

[49] Guo,S. Peter,L. (June 1994). A reconfigurable analog fuzzy logic controller, *IEEE World congress on computational intelligence. Proceeding of the Third IEEE conference on*, Vol. 1:pgs. 124-128.

[50] Jantzen,J. (15 May 1998). *Design Of Fuzzy Controllers*. Tech. report no 98-E 864.

[51] Pedrycz,W. (1993). *Fuzzy Control and Fuzzy Systems*, Third Avenue, New York: Wiley.

[52] Xinhong,Z. Zongxian,W. Zhengda,Y. (2007). *Intelligent Solar Tracking Control System Implemented on an FPGA*, Outstanding Designs.

[53] Yi,S. Chung,J. (1993). Identification of Fuzzy Relational Model and its applications to Control, *Fuzzy Sets and Systems*, 59, 1, 25- 33,

[54] Takagi,T. and Sugeno,M. (June 1985). Fuzzy identification of systems and its applications to modeling and control, *IEEE Transactions on Systems, Man and Cybernetic*, vol. 15, pp. 116-132.

[55] El Khateb,A. (2007). *OPTIMIZED HYBRID FUZZY FED PID CONTROL OF NONLINEAR SYSTEMS*. Master of Science in Electrical Engineering.

[56] Shull,N. (2006). *PROJECT EVALUATION USING FUZZY LOGIC AND RISK ANALYSIS TECHNIQUES*. Master of Science in Industrial Engineering.

[57] Moratori,P. Cruz,A. Lima,C. (July 2006). Comparing Robustness and Performance of Hybrid and Non-Hybrid Fuzzy Controllers aimed to guide a simulated robot. Proceedings of *IEEE International Conference on Fuzzy Systems*, pp. 1126-1132.

[58] Tulunay,Y. Senalp,E. Dorman,L. Tulunay,E. Mentes,S. & Akcan,M. (2008). *A fuzzy neural network model to forecast the percent cloud coverage and cloud top temperature maps*, Ann. Geophys., 26, 3945–3954.

[59] Yilmaz,M. Arslan,E. *Application of Fuzzy Logic Theory to Geoid Height Determination*, Harran and Istanbul, Turkey Universities .

[60] MathWorks. (2004). *Fuzzy Logic Toolbox For Use with MATLAB,* The MathWorks.

[61] Wongsuwarn,H. & Laowattana,D. (2006). *Neuro-Fuzzy Algorithm for a Biped Robotic System* , World Academy of Science, Engineering and Technology 15.

[62] Lee,C.C. (1990). Fuzzy Logic in Control Systems: Fuzzy Logic Controller, Part II. *IEEE Transactions on Systems, Man and Cybernetics*, 20(2):419--435.

[63] Leekwijck,W. Kerre,E. (1999). *Defuzzi_cation: criteria and classi_cation*, Fuzzy Sets and Systems 108 159-178.

[64] Takagi T. and Sugeno,M. (July 1983). Derivation of fuzzy control rules from human operator's control actions, *Proc. of the IFAC Symp. on Fuzzy Information, Knowledge Representation and Decision Analysis*, p. 55-60.

[65] Abdelati,M. (2007). *Modern Automation Systems*.

[66] (n.n). (n.d.). In *Wikipedia*. Retrieved June, 2010, from

http://en.wikipedia.org/wiki/Application-specific_integrated_circuit#cite_note-1

[67] (n.n). (n.d.). In *Wikipedia*. Retrieved June, 2010, from

http://www.fpgacentral.com/docs/fpga-tutorial/history-programmable-logic

[68] Xilinx Company. (June 12, 2006). *Programmable Logic Design, Technical data, Quick Start Handbook.*

[69] (n.n). (n.d.). In *Wikipedia*. Retrieved June, 2010, from

http://en.wikipedia.org/wiki/File:Programmable_Logic_Device.svg

[70] Jastrzebski,R. Laakkonen,O. Rauma,K. Luukko,J. Saren,H. Pyhonen,O. (June 28-30 2009). "Real-Time Emulation of Induction Motor in FPGA using Floating Point Representation. *Proceedings of the IASTED International Conference on Applied Simulation and Modelling*, Rhodes, Greece, pp. 226-231.

[71] Birkner,J. al,E. (n.d). A very-high-speed field-programmable gate array using metaltometal antifuse programmable elements, *Microelectronics Journal*, v. 23, pp. 561-568.

[72] Cirstea,M.N. Dinu,A. Khor,J.G. McCormick,M. (2002). *Neural and Fuzzy Logic Control of Drives and Power Systems*, Linacre House, Jordan Hill, Oxford OX2 8DP, First published.

[73] Freeman,R. (Dec. 1988). User-programmable gate arrays. *IEEE Spectrum*, vol. 25, Issue: 13,pp. 32 - 35.

[74] Pellerin,D. Thibault,S. (April 22, 2005) *Practical FPGA Programming in C*, Prentice Hall PTR,

[75] (n.n). (n.d.). In *Wikipedia*. Retrieved 2010, from http://en.wikipedia.org/wiki/File:Switch_box.svg.

[76] (n.n). (n.d.). In *Wikipedia*. Retrieved June, 2010, from http://en.wikipedia.org/wiki/File:Logic_block2.svg.

[77] Xilinx. (January 2010). *Xilinx Virtex-6 Family Overview*.

[78] Eichenlaub,S. & van,A. (Sept.1994). Getting the best from an FPGA via logic synthesis, *Conf. Record WESCON/94. 'Idea/Microelectronics*'., pp 587 - 591, 27-29.

[79] Venkateswaran,R. and Mazumder,P. (1994). A survey of da techniques for pld and fpga based systems. *Integration, the VLSI Journal*, 17:191-240.

[80] Salapura,V. & Hamann,V. (Sept. 1996). Implementing fuzzy control systems using VHDL and statecharts, *Proc. of Design Automation Conf., 1996, with EURO-VHDL '96 and Exhibition*, EURO-DAC '96, European, pp. 53 - 58, 16-20.

[81] Xilinx, Inc.(June 19, 2008). *Spartan-3A/3AN FPGA Starter Kit Board User Guide*.

Appendices

1.1 Appendix A

The key features of the Spartan-3AN Starter Kit:

Evaluation Board Specification	
Parameter	Description
Xilinx FPGA Device	
On-board Memory	4MBit Xilinx Platform Flash configuration PROM 4MByte parallel NOR Flash (2) 16 Mbit SPI Serial Flash
DDR2 Support	64MByte (512Mbit), 32M x 16 data interface.
I/O Connectors	High-speed differential I/O • Receiver: 6 data channels or five data channels plus clock • Transmitter: 6 data channels or five data channels plus clock Supports multiple differential I/O standards (eg. LVDS, RSDS, TMDS, PPDS, mini-LVDS) Supports 24 single-ended I/O
Additional Connectors	2, six-pin expansion connectors for Peripheral Modules (sold separately) 100-pin Hirose FX2 expansion connector with up to 43 FPGA user I/O SMA connector for clock inputs and outputs
USB Connector	For programming using supplied USB cable
ADC / DAC	2 channel SPI-based ADC with programmable gain 4 channel SPI-based DAC
Serial Ports	2, nine-pin RS232 (DTE and DCE-style)
User Interface	PS/2 port for mouse or keyboard
Display Interface	VGA display port, 12-bit color Two-line, 16 character LCD display
Ethernet	10/100 Ethernet PHY (requires Ethernet MAC in FPGA)
External Oscillator	133MHz socketed external oscillator
Audio Interface	Stereo audio jack using digital I/O pins
Switches / Knobs	Rotary-encoder knob with push-button shaft Four slide switches Four push button switches
LED	Eight discrete LEDs
Power	5.0 V DC, 1A
RoHS Compliance	Yes
Board Size	6.7" x 7.3" x 1.2" (170mm x 185mm 30mm)
Starter Kit Part Number	HW-SPAR3AN-SK-UNI-G

1.2 Appendix B:

L298

DUAL FULL-BRIDGE DRIVER

Multiwatt15

PowerSO20

ORDERING NUMBERS : L298N (Multiwatt Vert.)
L298HN (Multiwatt Horiz.)
L298P (PowerSO20)

nection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

PIN FUNCTIONS (refer to the block diagram)

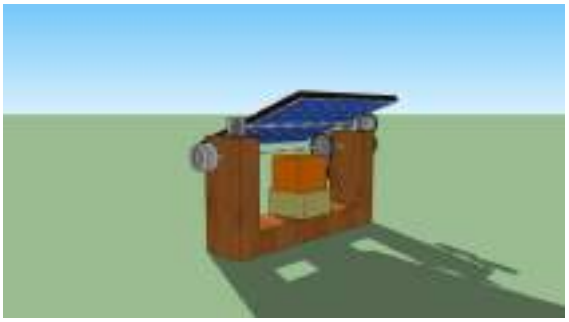
MW.15	PowerSO	Name	Function
1;15	2;19	Sense A; Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2;3	4;5	Out 1; Out 2	Outputs of the Bridge A; the current that flows through the load connected between these two pins is monitored at pin 1.
4	6	V_S	Supply Voltage for the Power Output Stages. A non-inductive 100nF capacitor must be connected between this pin and ground.
5;7	7;9	Input 1; Input 2	TTL Compatible Inputs of the Bridge A.
6;11	8;14	Enable A; Enable B	TTL Compatible Enable Input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
8	1,10,11,20	GND	Ground.
9	12	VSS	Supply Voltage for the Logic Blocks. A100nF capacitor must be connected between this pin and ground.
10; 12	13;15	Input 3; Input 4	TTL Compatible Inputs of the Bridge B.
13; 14	16;17	Out 3; Out 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15.
–	3;18	N.C.	Not Connected

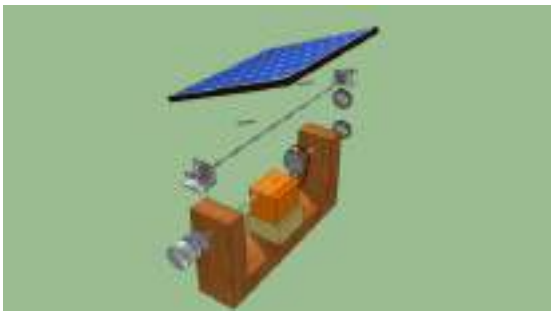
1.3 Appendix C:

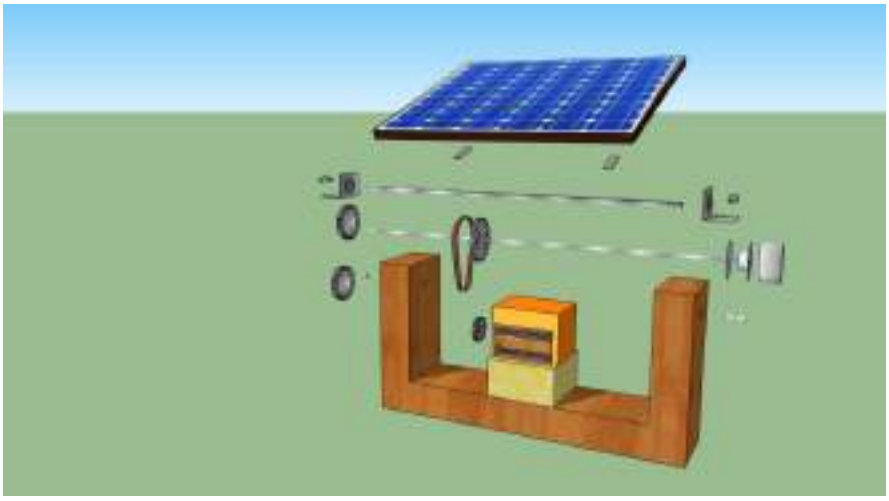
Sun tracker in 3D graphs represented in real dimensions using AutoCAD program .

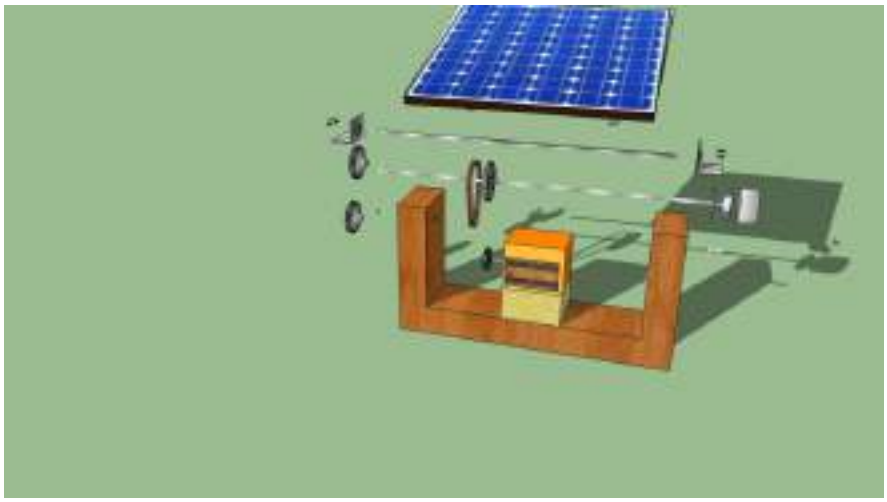


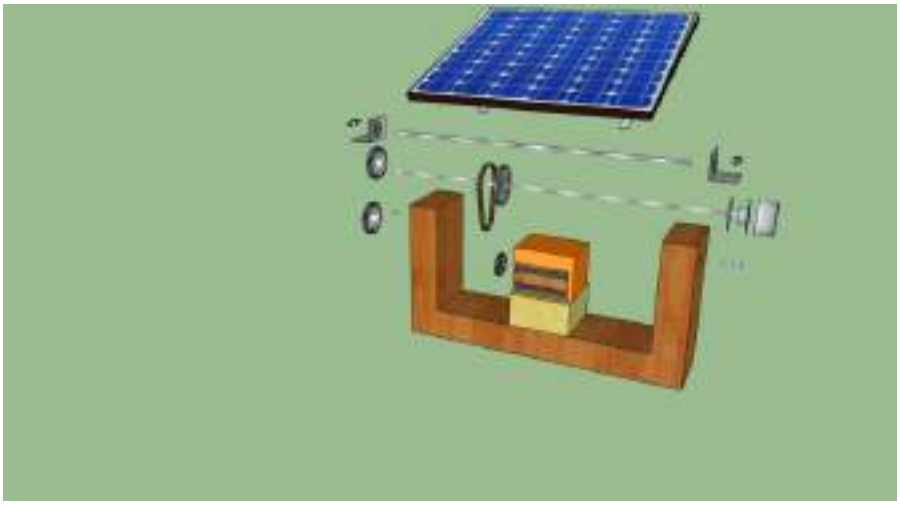










1.4 Appendix D:

```
--PWM VHDL code:
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity PWM is
    Port ( CLK50 : in std_logic;
           DB    : in std_logic_vector(7 downto 0);
           PWMout: out std_logic);
end PWM;

architecture Behavioral of PWM is
signal DBbuf   : std_logic_vector (15 downto 0);
signal counter : std_logic_vector (15 downto 0):="0000000000000000";

begin
DBbuf(15 downto 8) <= DB (7 downto 0);
DBbuf(7 downto 0) <= "00000000";

process (CLK50)
begin
if CLK50'event and CLK50='1' then
      if counter < "1111111111111110" then
         counter <= counter +1;
   else
         counter <= (others =>'0');
   end if;
      if DBbuf > counter then
            PWMout <='1';
   else
            PWMout <='0';
      end if;
end if;

end process;

end Behavioral;
```

1.5 Appendix E:

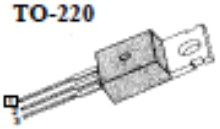
Advanced Power MOSFET **IRF740A**

FEATURES

- Avalanche Rugged Technology
- Rugged Gate Oxide Technology
- Lower Input Capacitance
- Improved Gate Charge
- Extended Safe Operating Area
- Lower Leakage Current: 10μA (Max.) @ V_{DS} = 400V
- Lower $R_{DS(ON)}$: 0.437Ω (Typ.)

BV_{DSS} = 400 V
$R_{DS(on)}$ = 0.55Ω
I_D = 10 A

TO-220



1.Gate 2. Drain 3. Source

Absolute Maximum Ratings

Symbol	Characteristic	Value	Units
V_{DSS}	Drain-to-Source Voltage	400	V
I_D	Continuous Drain Current (T_C=25°C)	10	A
	Continuous Drain Current (T_C=100°C)	6.3	
I_{DM}	Drain Current-Pulsed (1)	40	A
V_{GS}	Gate-to-Source Voltage	±30	V
E_{AS}	Single Pulsed Avalanche Energy (2)	457	mJ
I_{AR}	Avalanche Current (1)	10	A
E_{AR}	Repetitive Avalanche Energy (1)	13.4	mJ
dv/dt	Peak Diode Recovery dv/dt (3)	4.0	V/ns
P_D	Total Power Dissipation (T_C=25°C)	134	W
	Linear Derating Factor	1.08	W/°C
T_J , T_{STG}	Operating Junction and Storage Temperature Range	- 55 to +150	°C
T_L	Maximum Lead Temp. for Soldering Purposes, 1/8, from case for 5-seconds	300	

Thermal Resistance

Symbol	Characteristic	Typ.	Max.	Units
$R_{\theta JC}$	Junction-to-Case	–	0.93	°C/W
$R_{\theta CS}$	Case-to-Sink	0.5	–	
$R_{\theta JA}$	Junction-to-Ambient	–	62.5	

Rev. B

FAIRCHILD SEMICONDUCTOR™

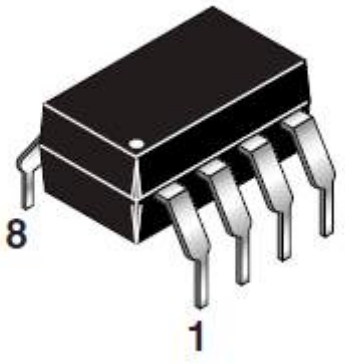
1.6 Appendix F:

HIGH SPEED-10 MBit/s LOGIC GATE OPTOCOUPLERS

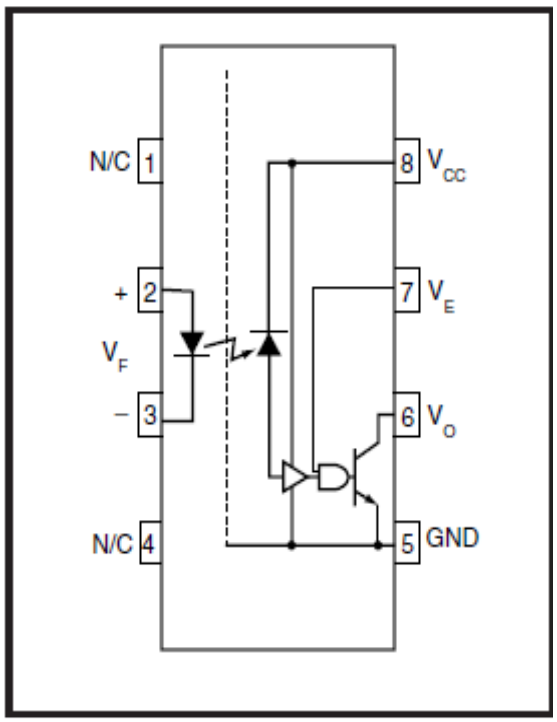
SINGLE-CHANNEL

6N137

DESCRIPTION



The 6N137, HCPL-2601/2611 single-channel optocouplers consist of a 850 nm AlGaAS LED, optically coupled to a very high speed integrated photodetector logic gate with a strobable output. This output features an open collector, thereby permitting wired OR outputs. The coupled parameters are guaranteed over the temperature range of -40°C to +85°C. A maximum input signal of 5 mA will provide a minimum output sink current of 13 mA (fan out of 8). An internal noise shield provides superior common mode rejection of typically 10 kV/µs. The HCPL- 2601 and HCPL- 2631 has a minimum CMR of 5 kV/µs. The HCPL-2611 has a minimum CMR of 10 kV/µs.



6N137

FEATURES

- Very high speed-10 MBit/s
- Superior CMR-10 kV/µs
- Double working voltage-480V
- Fan-out of 8 over -40°C to +85°C
- Logic gate output
- Strobable output
- Wired OR-open collector
- U.L. recognized (File # E90700)

APPLICATIONS

- Ground loop elimination
- LSTTL to TTL, LSTTL or 5-volt CMOS
- Line receiver, data transmission
- Data multiplexing
- Switching power supplies
- Pulse transformer replacement
- Computer-peripheral interface

1.7 Appendix G:

International
IOR Rectifier

Data Sheet No. PD60147-M

IR2110/IR2113

HIGH AND LOW SIDE DRIVER

Features

- Floating channel designed for bootstrap operation
  Fully operational to +500V or +600V
  Tolerant to negative transient voltage
  dV/dt immune
- Gate drive supply range from 10 to 20V
- Undervoltage lockout for both channels
- Separate logic supply range from 5 to 20V
  Logic and power ground ±5V offset
- CMOS Schmitt-triggered inputs with pull-down
- Cycle by cycle edge-triggered shutdown logic
- Matched propagation delay for both channels
- Outputs in phase with inputs

Product Summary

V_{OFFSET} (IR2110)	500V max.
(IR2113)	600V max.
I_O+/-	2A / 2A
V_{OUT}	10 - 20V
$t_{on/off}$ (typ.)	120 & 94 ns
Delay Matching	10 ns

Packages

14 Lead PDIP
IR2110/IR2113

16 Lead SOIC
IR2110S/IR2113S

Description

The IR2110/IR2113 are high voltage, high speed power MOSFET and IGBT drivers with independent high and low side referenced output channels. Proprietary HVIC and latch immune CMOS technologies enable ruggedized monolithic construction. Logic inputs are compatible with standard CMOS or LSTTL output. The output drivers feature a high pulse current buffer stage designed for minimum driver cross-conduction. Propagation delays are matched to simplify use in high frequency applications. The floating channel can be used to drive an N-channel power MOSFET or IGBT in the high side configuration which operates up to 500 or 600 volts.

Typical Connection