DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

REMOTE MONITORING OF PHOTOVOLTAIC SYSTEMS

by

Tunca KÖKLÜ

March, 2017

İZMİR

REMOTE MONITORING OF PHOTOVOLTAIC SYSTEMS

A Thesis Submitted to the
Graduate School of Natural And Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science in
Electrical and Electronics Engineering Program

by
Tunca KÖKLÜ

March, 2017
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"REMOTE MONITORING OF PHOTOVOLTAIC SYSTEMS"** completed by **TUNCA KÖKLÜ** under supervision of **ASSOC. PROF. DR. SELÇUK KILINÇ** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

S. ey

Assoc. Prof. Dr. Selçuk KILINÇ

Supervisor

Assoc. Prof. Dr. Engin Karatepe (Jury Member)

Asst. Prof. Dr. Mümtaz YILMAZ (Jury Member)

Prof.Dr. Emine İlknur CÖCEN
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

First of all, I would like to acknowledge and express my gratitude to my parents for hearten me to follow a path of passion, joy and purpose in every areas of life.

Also importantly, I would like to thank Assoc. Prof. Dr. Selçuk Kılınç and Asst. Prof. Dr. Özgür Tamer for sharing their wisdom. Finally, I would like to thank Anıl Köklü, Türker Türkoral and Recep Öztürk to their friendship and support.

Tunca KÖKLÜ

REMOTE MONITORING OF PHOTOVOLTAIC SYSTEMS

ABSTRACT

Remote monitoring of photovoltaic systems is crucial part of the established solar systems for users and researchers. The performance and working condition of each part existing in photovoltaic systems should be observable in order to run them in optimum condition and efficiency. In this thesis, a design has been realized for remote monitoring of a 1 kW photovoltaic system. With this design, it is aimed to transfer measured information at the energy station to the center station using Message Passing Interface. Input and output voltages and currents of maximum power point tracker, voltage of each battery, temperature, irradiation, dust density, humidity and efficiency of maximum power point tracker can be observed remotely using this design. Some of the measurements are taken by sensors or circuits, which gives analog output, and microcontroller converts them in to digital signal. Converted information have been sent using serial channel from the microcontroller to the single board computer. The sensor, which gives digital output, has been connected to the single board computer. All these information is sent to center station using Message Passing Interface. Incoming information have been disposed to related tables of database in the center station. Specifically, a cluster of single board computers has been used in the design. This cluster is composed of a center single board computer and remote station single board computers as many as the photovoltaic system count. Measurements are broadcasted over internet and viewed on a graphical user interface using display at the center station. Warning unit of graphical user interface has been designed based on parallel processing principle. Stored measurements are divided to all processors and every piece of data is executed by single board computer simultaneously. Measurement data, which are beyond the limitations, are described as mismatched results. Results can be observed with user interfaces.

Keywords: Photovoltaic systems, remote monitoring, single board computers, parallel processing.

FOTOVOLTAİK SİSTEMLERİN UZAKTAN İZLENMESİ

ÖZ

Fotovoltaik sistemlerin uzaktan izlenmesi kullanıcılar ve araştırmacılar açısından kurulmuş sistemin önemli bir parçası olarak görülmektedir. Sistemlerde bulunan her parçanın performans ve çalışma koşulları, bu parçaların en uygun durum ve kazançta çalışması için gözlenebilir olmalıdır. Bu çalışma 1 kW gücündeki bir fotovoltaik sistemin uzaktan izlenmesi için tasarlanmıştır. Çalışmada enerji istasyonunda ölçülen verileri merkez istasyona mesaj aktarma arayüzünden göndermek hedeflenmiştir. Maksimum güç noktası izleyici giriş ve çıkış akım ve voltajları, akü gerilimleri, ortamın sıcaklığı, ışınım, toz yoğunluğu, nem bilgileri ve maksimum güç noktası izleyici verimliliği bu sistemle uzaktan gözlemlenebilir. Ölçümlerden bazıları analog çıkış veren sensör veya devrelerden alınmıştır ve mikrokontrolör bu ölçümleri dijital sinyale çevirir. Dijital sinyale çevrilen bilgiler seri kanal kullanılarak ölçüm işlemcisinden tek kart bilgisayara gönderilir. Dijital çıkış veren sensörler ise uzak istasyonda bulunan tek kart bilgisayarlara bağlanmıştır. Tüm bilgiler merkez istasyonda bulunan tek kart bilgisayara mesaj aktarma arayüzü ile gönderilir. Gelen bilgiler merkez istasyondaki ilgili veritabanı tablosuna aktarılır. Tasarımda bu özellikleriyle bir tek kart bilgisayar kümesi kullanılmıştır. Bu küme, bir merkezi tek kart bilgisayar ve gözlemlenecek uzak fotovoltaik istasyon sayısı kadar tek kart bilgisayar içermektedir. Ölçüm bilgileri merkezde konumlandırılmış tek kart bilgisayar üzerinden eşzamanlı olarak internette yayınlanır ve merkezde bulunan ekrandan, kullanıcı arayüzü kullanılarak incelenebilir. Görsel arayüzün uyarı birimi paralel işleme ilkesi temelinde tasarlanmıştır. Depolanmış ölçüm bilgileri bütün işlemçilere bölünür ve bölünmüş bilgiler ilgili işlemcide eşzamanlı olarak incelenir. Önceden belirlenmiş limitlerin dışında olan ölçümler uyumsuz sonuçlar olarak tanımlanır. Sonuçlar kullanıcı arayüzlerinden incelenebilir.

Anahtar kelimeler: Fotovoltaik sistemler, uzaktan izleme, tek kart bilgisayarlar, paralel işleme.

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE
INTRODUCTION

World energy consumption has rapidly raised every year without considering the depletion of fossil fuel and taking any precautions for the future. Statistics about energy are obtained periodically each year and rate of all energy produced from all known sources was included in the statistical results to point out excessive energy consumed throughout a year (Energy information administration statistic, 2016). A great amount of energy naturally exists, such as sun and wind power. However, these type of energy should be transformed to practical usage form. Energy production can be investigated under two major titles such as renewable and non-renewable methods. Non-renewable energy is produced from limited sources like fossils. Renewable energy can be generated from continuous cycle of nature and they can be also replaced by earth natural cycles. Energy production, transformation and usage methods are common issues between nature and humankind because of their directly or indirectly outcomes like carbon footprint and pollution of nature. Outputs of these kind of issues directly affect on nature and most of them are considered as irreversible results.

Under all these facts, the demand for renewable energy is increasing and therefore, alternative energy production methods are being discovered day by day to save our fuFture. Global electricity production statistics and total renewable electricity production, which include only wind and solar electricity production per each country, can be analysed using statistical results. Developed countries renewable electricity production percentage is much higher than developing countries according to this statistical results (Renewable report, 2016).

The instruments used for production of renewable energy are becoming cheaper with the highly advanced progress of energy production technology. Therefore, fund investment tendency to energy has been drastically increased to enhance the renewable methods in last decades. Global electricity production using renewable methods were investigated by US Energy Information Administration every year. According to the most recent and approved statistical results (2000-2012), every continent has given enormous attention in order to increase green energy production percentage for their future (Energy information administration statistic, 2016).

Energy demand in Turkey has been continuously raising but increase in the energy production does not compensate with the consumption level and hence, importation of energy increases. Therefore, energy importation is rendered depending on other countries. General Directorate of Electrical Power Resources presents a map, which contains irradiation of sun in Turkey for each city. South of Turkey have much higher irradiation level than north so the investments are mostly located south part of Turkey (Solar energy in Turkey, 2016). Especially areas, which does not cultivated, have so much potential to produce energy from green source like biomass, solar and wind. Ministry of Energy and Natural Resources supports to investors and guarantee to buy produced energy with constant price (Turkey renewable law, 2016).

Identification of solar system performance's variation and investigation of the reason of these variation is crucially significant for users and researchers. Related data of each component has to be observable to intervene with the device during unexpected measurement. Therefore, remote monitoring of the photovoltaic systems is an important issue. Remote monitoring of solar systems consists of three main blocks as; measurement, observation and communication units. Measurement of essential quantities, measurement of photovoltaic related natural conditions can be ordered as the sub-divisions of measurement unit. Transmission of measured quantities are major part of remote monitoring of solar systems. Distance between center and remote station and location of every stations are important parameters to decide communication type and they also directly affect cost, speed, reliability and system requirements. Therefore, system requirements should be carefully examined before making decision about communication types.

Classification, storage, report and user interfaces can be defined as sub-divisions of the observation unit. Every measurement is classified by considering environment or photovoltaic plants related readings. Thus, the storage device becomes directly effective on cost of the system and investigation of data. Feedback report can be easily visualized to spontaneously treat the device by assessing its user friendly control algorithm. Report system, based on user essential requirements, creates possibility to examine working conditions and modifies the broken components by giving instant response of the operator. Analysis of desired quantities with appropriate intervals allows us to compare and study each measured data sets using feedback report information. Therefore, components ,which can not properly fulfil the function during the operation time, can be instantly replaced with new one.

User interfaces can be identified as a human-machine interaction where human requires to communicate with a device or machine. Processed data from source can be investigated by user interface permit to prevent user, who does not have enough skills to interact with machine language, intervention of main code blocks. Interface should let users observe every data. Otherwise, measured quantities have been useless. Therefore, data, which is directly related to system, should be presented with effective illustration methods like graphs and tables by user interfaces.

In the literature, examination of photovoltaic system is accomplished by controlling and monitoring it. Photovoltaic system's maximum power point tracker (MPPT) device and remote monitoring of related data designs were established by researchers (Xiaoli & Daoe, 2011). MPPT was controlled by DC - DC converter and output was modified over microcontroller. Zigbee method is used for communication. Other two studies were only focused on monitoring the measured data. Communication methods of the were was established over Zigbee (Rashidi et al., 2011; Bagnasco et al., 2012).

In the World Solar Congress hold in 2013, a remote monitoring design was presented. Global system for mobile communications (GSM) voice channel was used to accomplish communication between the remote station and the center station. Remote stations were located into rural areas. Various experiments were conducted between 50-4000 Hz data rate and the applied voltage was changing from 0 to 6 V. They succeeded to send the data in the frequency range 300-3300 Hz and 4.5 V amplitude with very small error rate. GSM voice channel was accepted an appropriate solution for applications, which are focused on remote stations in rural areas in this design. There was also comparison of communication as general packet radio service (GPRS)-GSM communication and researchers claimed that 5% of solar plant's cost is enough to build this communication system. It was well promising that this technology can be implemented in rural districts to reduce cost and energy consumption (Tejwani et al., 2013).

In 2014, another study was accomplished using wireless sensor network (WSN) and radio frequencies are used for communication purpose considering sensitivity of device in terms of frequency interval. Dust, humidity, temperature, light and current sensors have been used to create the WSN and graphical user interface (GUI) was designed by researchers. Continuous monitoring of solar plants was considered as the most essential requirements for economic success but it still needs many further steps to enhance design in order to exclude minor defects.(Al-Dahoud et al., 2014).

In 2004, database system was designed for remote monitoring. The readings are stored in external disk and they were immediately sent to related database table. Focusing on creating effective statistical outcomes of real-time photovoltaic related measurement was the key concept of the design. Transmission Control Protocol/Internet Protocol (TCP/IP) communication protocol was used to establish communication and intervention to database remotely(Papadakis et al., 2005).

Researchers generally focused on power plants located in rural areas and hence cost of communication modules were added to the system cost. However, more effective and cheaper solution is establishment of the system in the urban area. It should measure the related quantities and communicate between remote center stations for low powered solar plants located near urban areas. Remote monitoring of solar systems can be branched on three titles which are (Al-Dahoud et al., 2014) computer (remote station) to computer (center station), embedded (remote station) to computer (center station), embedded (remote station) to embedded (center station).

Ethernet can be used for network connection of center station. Cable cost and location of photovoltaic plants are constituted an impediment for the solution (Fang &

Wang, 2011). Dial up network can be practiced to data transmission. However, novel technologies which are third generation (3G), ADSL can not fulfil cost and reliability conditions. GSM modem, dial up network and ethernet are widely preferred for different data transmission types of embedded remote monitoring designs.

Ethernet and dial up network has certain drawbacks when it is used for solar plants applications because of location of systems or devices. GSM modem can be used in two ways through GPRS or short message service (SMS). These two transmission types are directly depend on service provider. GPRS, which is a packet oriented service, is preferred in area ,where wireless network is available. Text transmission correlated 160 alfa-numeric characters is called SMS. Delays and rarely fails occurs (Tejwani et al., 2013) in transmission. Design's data transmission type decides itself after analyzing initial requirements given at the beginning of measurements.

In this thesis, design was completed for the remote monitoring of a 1 kW photovoltaic system. Four 250 W photovoltaic panel were connected each other in this system. They were paired to connect series to each other. Thus, these pairs were connected parallel to each other. Remote monitoring of the photovoltaic system design was realized in embedded remote station to embedded center type. Solar plant located near urban area was chosen as measurement region.

Outputs of photovoltaic system under investigations are directly connected to sensors or designed measurement circuits. Measurements, which are taken from analog sensors, connected microcontroller to convert them digital. Input and output voltage and current of MPPT, voltage of each battery, irradiation, temperature, dust density and humidity are measured using this method. These measured quantities sent over universal asynchronous receiver-transmitter (UART) pins, which are suitably crossed between single board computer and microcontroller. Temperature and humidity sensors were connected to the single board computer, directly. All measurements are collected by a single board computer in remote station.

Data transmission between center and remote station is accomplished over ethernet using Message Passing Interface (MPI) program. MPI is mostly used to build cluster computer. It allows to execute functions ,simultaneously without missing data. It is the first reason why the program was chosen. In 2010, Iridispi cluster computer is constructed by researchers and it achieved to enter top 500 cluster computer list (Cox et al., 2014). On the other hand, data transmission between the members of cluster can be coded in special syntax in different programs like Python,C,C++ and Fortran. Execution time can be reduced using this concept (Köklü et al., 2015) and it is the second reason why the program is chosen.

Measurements, which are taken from the remote stations, are sent by MPI to center station as mentioned. Every data stored related database's table to broadcast and visualize them over internet and center station's user interface. Center single board computer is responsible for communication, storage, classification, measurement timing, user interfaces and report. MPI is widely used standardized message passing system for parallel communication. Parallel processing is used for analysis of related measurements.

Effectiveness of design was identified as storage and classification of the measured data in observation and examination sections. Structured query language (MYSQL) database was used to achieve the goal. Every remote station has its own table and every data has its own column in the center station.

Timing of measurement is identified by center single board computer because of the architecture of the parallel processing. Master and slave relation is occurred between remote and center station. All stations have partially same code in same location/folder and this code must include duty of every station. This program is executed with master station order and timing improvement of this method was proven (Chorley & Walker, 2010).

User interfaces are only way to present the measured data to users since they do not understand machine language. Seven inch display which was located on the center station was used to create a GUI in the design. Effective illustration methods like graph is used to present the data. Website is also used to visualize the data. Login protected website, which consists of two major parts: user and manager, is also preferred to observe measurement. Login authority is given from designer to users and other managers. Users and managers can be deleted by manager after the authority is declared. Information of measurements and graphs can be seen and only information of measurements can be downloaded by all members.

GUI is located in the center and measured data can be observed by users on graphs and information boxes. In report screen, previous data between the identified interval can be examined and downloaded. Measured information can be analyzed using warning unit in GUI. This unit works with parallel processing principle. Informations, which will be examined, is distributed equally to all available processors. Distributed data is executed with same instruction set.

The results are controlled by determining the upper and lower limits. If the results are not in the limit range, it is automatically sent back to the center station for self-correction process. Limits are identified using datasheets, specifications of related devices and measurement information. Datasheet of batteries has specification about maximum and minimum temperature. Every cycle of battery charge is observed and minimum and maximum voltage value for each batteries is identified using them. Output of the MPPT is connected to a load to measure current in both side of the MPPT. Current consumption rate of the load is also known and they are used as limitations of currents. On the other hand, output voltage of the MPPT can be determined by checking the voltage requirements of the load too. Thus, output voltage of MPPT limitations are identified using this information and these limits should be changed according to established system. All these designs, codes and used components can be seen on website and can be used by related users. They are all open source design.

This thesis is organized as follows. Second chapter gives theoretical definitions of used programs, processors and techniques. Each program and the reason of using it introduced. Therefore, processors, which are used in the design, is presented in this chapter. Every part of the design is explained according to order of design in designed remote monitoring system chapter. Chapter four involves the related results and measurements. GUI and website usage informations with figures are shared in this chapter. Concluding remarks and points are presented in the fifth chapter.

CHAPTER TWO
BACKGROUND INFORMATION

Remote monitoring of photovoltaic system is crucial for users. Thus, every related parameter must be observed using remote monitoring device. Photovoltaic plant, which will be observed remotely, must be carefully analyzed. First section of this chapter includes connection between photovoltaic panels. Both series and parallel connection are used in the photovoltaic plant, which will be observed remotely. Therefore, connection types are explained. On the other hand, output voltage and current of the photovoltaic plant can be calculated using connection of panels. However, output voltage and current of photovoltaic plant must be measured to compare theoretic and real outcomes. Second section of the chapter includes hardware(processors) ,which are used to measure related quantities, are explained. Measured parameters must be sent to center station. Therefore, communication between remote and center station is accomplished using MPI.

MPI is message passing interface program, which was designed for parallel communication and simultaneous execution order can be sent every remote station using MPI program. However, hardware of remote monitoring system must be analyzed according to instruction and data pool to understand system architecture. Thus, types of the architecture of an hardware are created the third section of the chapter. On the other hand, parallel processing is used to control past data in designed system. Parallel performance of the system must be defined and calculated using universal parallelism rule. Thus, equations, which are used to define parallel performance, is explained in last part of the section three.

Remote Monitoring of the system is accomplished using website. In the last section of the chapter used programs and their abilities is explained. Finally, This chapter includes the information about programs, used hardware and theoretical aspects in this thesis. Examination of necessities were done before the decision of processors and programs. Chosen programs are free and processors are easy to find and use.

2.1 Photovoltaic Panels

Measured quantities of photovoltaic systems are determined considering necessities to run the system with high efficiency. Detailed examination of photovoltaic systems is required for identification of parameters. Photovoltaic panels are designed to absorb sun lights and convert the energy from the sun to electric energy. Photovoltaic cell is created by a P-N junction and the silicon based layers are composed of N-P type material. Sun energy absorbed by photovoltaic cells is converted to voltage between these junctions. Direct current is generated by constant one direction moves. Outer layer of photovoltaic cells consists of protections and coatings to avoid undesired effects such as reflections. Layers of photovoltaic panel is shown in Figure 2.1.

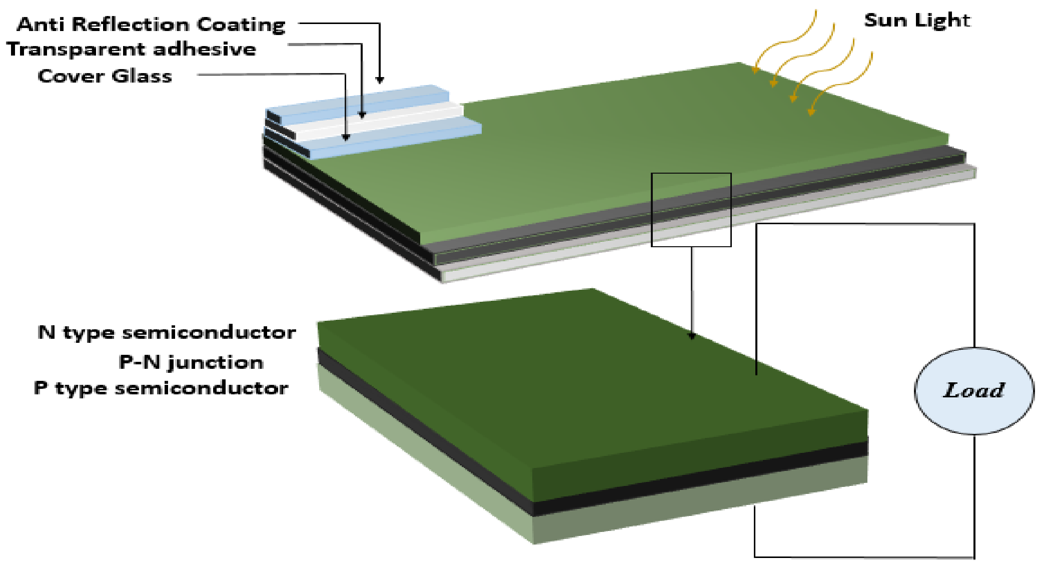


Figure 2.1 Layers of a photovoltaic panel

Photovoltaic arrays are created by connected photovoltaic panels. Connection types like series, parallel or combination of them can be chosen to reach aimed output voltage and current.

2.1.1 Series Connection

Photovoltaic panels are connected serial to obtain high voltage. It is well-known fact that series connected panels increase voltage level. Positive terminal of first solar panel is connected to next solar panel's negative terminal and principle is the same for other panels. Series connected panels can be seen in Figure 2.2.

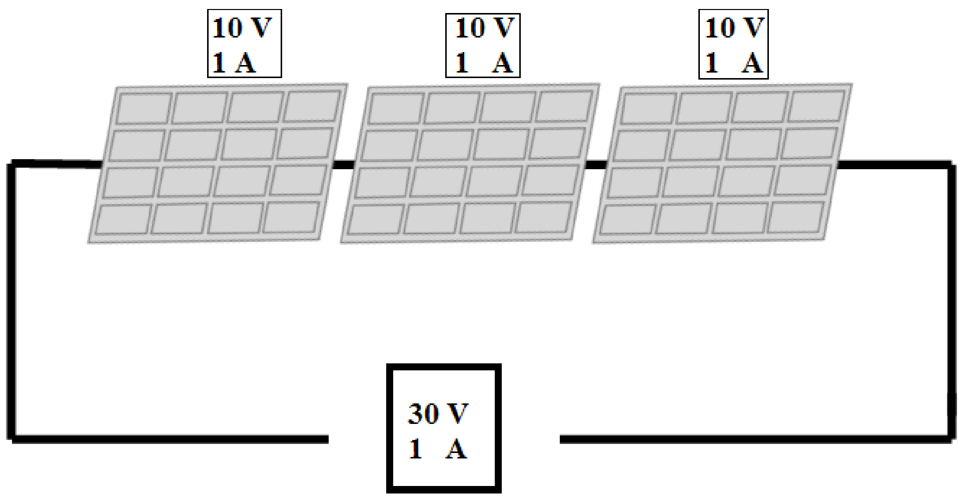


Figure 2.2 Series connected photovoltaic panels

Series connection with different current-voltage characteristic can be investigated in two parts. Series connection with different voltage and same current does not cause a problem. The total voltage produced by sun energy is the sum of the all voltages. However, different current limits the output current as the lowest panel current even if the panel voltages are same or different. Therefore, photovoltaic array power can not be used with high efficiency.

2.1.2 Parallel Connection

Solar panels are connected parallel to each other in order to obtain high current as much as loads or batteries need and resist. Positive and negative terminals of panels are connected to each other. Solar panels operated with varying powers can connect parallel if the voltage level keep same. However if the voltage of panels are different, they should not be connected parallel because panel with low voltage acts like load and it consumes energy and current, instead of producing. Parallel connected panels can be seen in Figure 2.3.

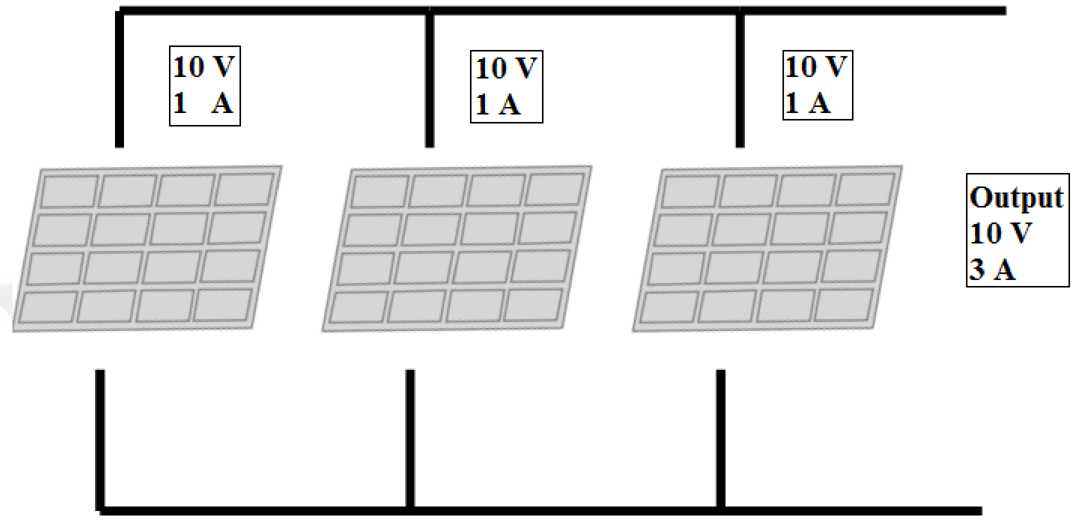


Figure 2.3 Parallel connected photovoltaic panels

2.1.3 Combination of Series and Parallel Connection

The photovoltaic arrays connected only series or only parallel can not be used for every load. It may be dangerous for charge controller in the presence of high current. Current and voltage requirements of load identify the system connection type. Voltage and current can be simultaneously increased using both series and parallel connection. Schematic of series and parallel combination is shown in the Figure 2.4 and this figure is also a part of the designed system, which is studied in this thesis.

2.2 Hardware

Input and output voltage and current of MPPT, voltage of each battery, irradiation, temperature, dust density and humidity are measured. Outputs of photovoltaic system

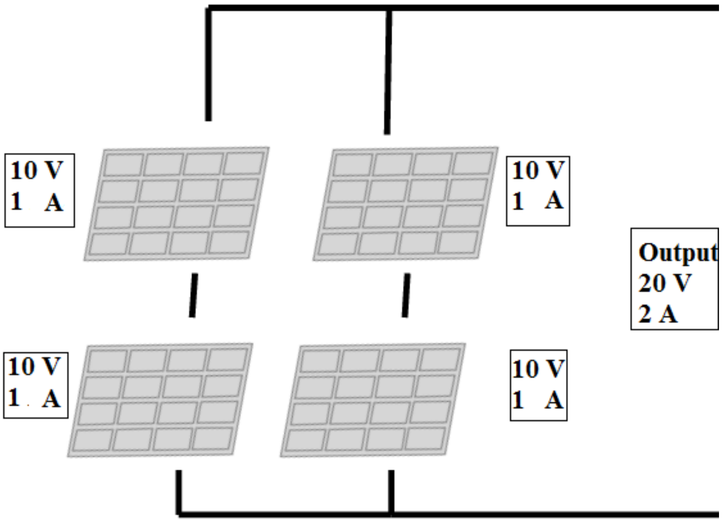


Figure 2.4 Series and parallel combination connected solar panels

are directly connected to sensors or designed measurement circuits. Measurements, which are taken from analog sensors, connected microcontroller to convert them digital. Information, which are taken from analog sensors, is directly sent single board computer using serial channel. Single board computer located in remote station collects both digital and analog measurements and sent center station single board computer. PIC was used as a microcontroller and Raspberry pi is used as a single board computer.

2.2.1 Raspberry Pi

Raspberry Pi is small sized computer. Keyboard, mouse and monitor are required in order to run and use it like personal computer. Python and C compilers are loaded in every operating system (OS). Various types and versions can be found between 25-35 dollar in global distributors. It can be used as a receipt printer, surveillance systems processors or any project, which needs embedded system/systems. Every version or type of cards have general purpose inputs outputs (GPIO), USB, ethernet and many other outputs and inputs. OS can be chosen from official website of the Raspberry Pi from list, which includes of Fedora, Raspbian and only for Raspberry Pi 2 Windows-10 is available. OS can be load using SD or micro-SD card according to version of Raspberry Pi. Raspbian is used to accomplish the work in this study.

Raspberry Pi GPIO has two 5 V, a 3.3 V supply pins and four ground pins. Serial peripheral interface (SPI), UART, interface to communicate (I2C) pins can be used for communication. There are also so many pins to use in projects as outputs to control connected components or as inputs to read incomes except analog one. Raspberry pi 1 model A and B GPIOs can be seen Figure 2.5.

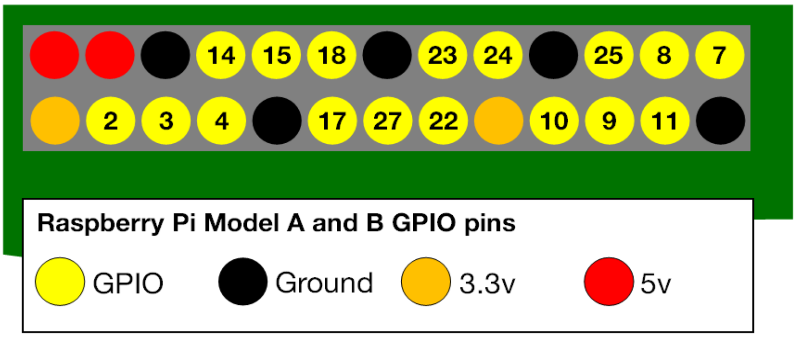


Figure 2.5 Raspberry Pi GPIOs list and their assigned duties (Raspberrypi GPIO, 2016)

2.2.2 Peripheral Interface Controller (PIC)

PIC 18F4580 is a microcontroller made by Microchip. Stability, reliability and wide range usage capability are considered advantages to chose it. Eleven pins analog to digital converter can be used to convert information taken by analog outputted sensors or circuits. SPI, UART, I2C pins can be used for communication to other devices. In study, PIC is used to analog to digital conversion because used single board computer does not have analog inputs. Furthermore, communication with Raspberry Pi and PIC is pretty simple. Serial channel is used for communication between PIC and Raspberry Pi. Voltage level of the processors are different so voltage levels must be adjusted using voltage divider before communication. Serial communication is succeed without any missing data. GPIO pins of PIC can be seen in Figure 2.6.

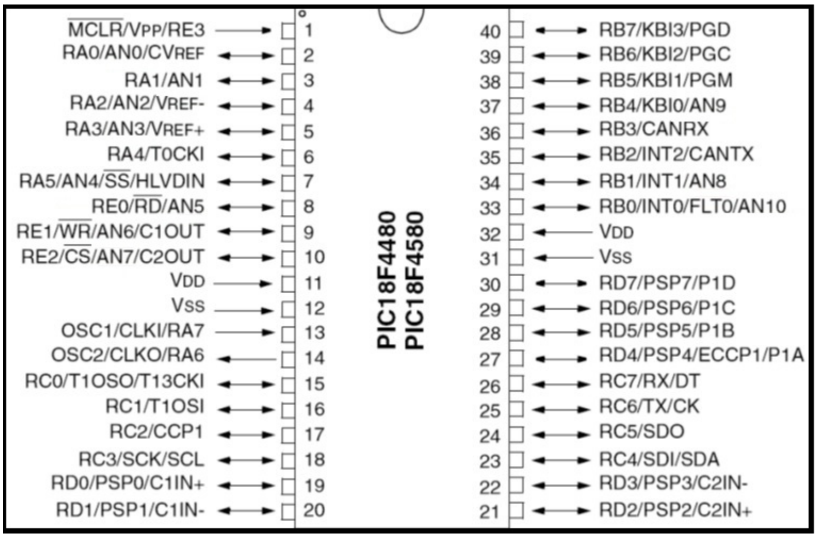


Figure 2.6 PIC 18F4580 pin list and their assigned duties

2.3 Computer Architecture

In the designed remote monitoring system, single board computers are used and some parts of the thesis are designed based on parallel processing. Processing of data is executed by multi-processors in parallel processing. Using this method, many calculations can be done simultaneously in different processors. However, in the sense of embedded system's parallel processing, the code, which is written to apply the method, is one of the most important part for performance examination. It also directly affects the speed up of a process.

Parts of some processes are executed simultaneously by using more than one processors with less time in the concept of parallel processing. Due to working with big data sets, massive operations have become an ordinary and common issue in academia and business environment with technological improvements. Massively strong computer-processors are required especially for researchers dealing with big data sets like weather and quantum fields.

On the other hand, users could not respond by hardware. Parallel processing was discovered as a solution and applied by researchers in last decades and tens of thousands processors are being available for ten years in parallel computing (Gabriel et al., 2004). In this point of view, desired process is completed more than one processors. Hardware of device is one of the most important parameter of parallel processing. Hardware of the system can be categorized by Flynn taxonomy and Von Neumann architecture (Backus, 1978; Flynn, 1972).

2.3.1 Von Neumann Architecture

Von Neumann architecture is basically defined as a digital computer which consists of processing unit, control unit, memory and input-output. Processing unit (CPU) is composed of arithmetic logic unit (ALU) and control unit. Execution order is given by control unit and execution is handled by ALU. Data and instructions are both stored in memory. This architecture can be seen in Figure 2.7.

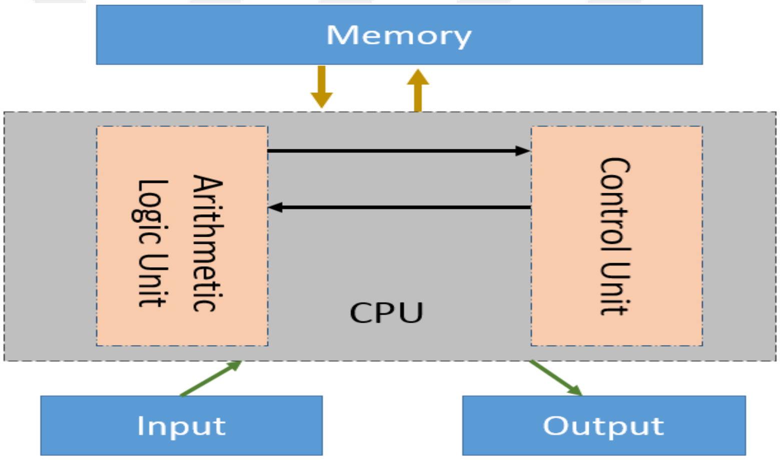


Figure 2.7 Computer parts according to Von Neumann architecture

2.3.2 Flynn Taxonomy

If available hardware can not run the desired process, parallel processing can be used in the systems. Flynn taxonomy can be investigated under four groups. Single Instruction stream, Single Data stream (SSID), Single Instruction stream, Multiple Data streams (SIMD), Multiple Instruction streams, Single Data stream (MISD), Multiple Instruction streams, Multiple Data streams (MIMD) are the branches of this taxonomy (Flynn, 1972).

2.3.2.1 *SSID*

A data set is executed from a processor. The method of execution is called as single instruction stream or single data stream. There is no parallelism in this method. Single command is executed at a single time and when this execution proceeds another command can not be processed. Method can be seen in Figure 2.8.

2.3.2.2 *SIMD*

In this part, instruction set is only one but data set and processors are multiple. Data sets are executed with one instruction set. Array processors are example of this kind of systems. In this study, built communication is acted like SIMD. Architecture can be examined in Figure 2.8.

2.3.2.3 *MISD*

This kind of systems have more than one processors and instruction set but data flow is organized by dataset. Same data set can be executed with different instructions so it can be useful to check different method's result. The detailed of the method is illustrated in the Figure 2.8.

2.3.2.4 *MIMD*

Processors can be operated asynchronously using multiple instruction and multiple data set. More than one SSID device act like MIMD. This architecture can be investigated under two branches as shared or distributed memory. Architecture of MMID can be seen in Figure 2.8.

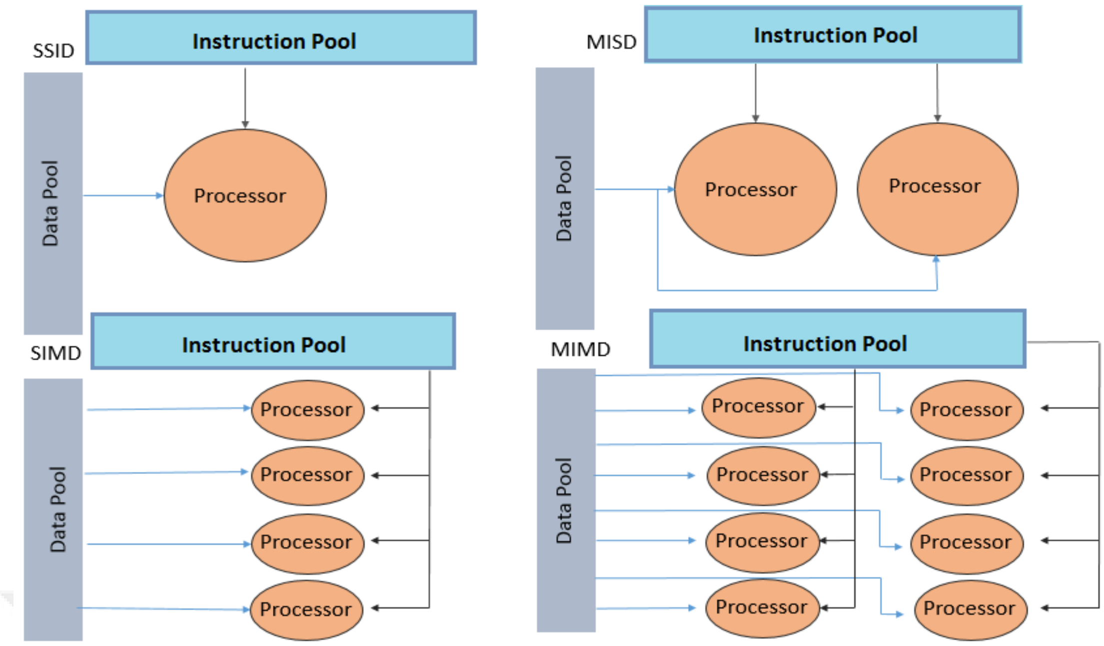


Figure 2.8 Flynn taxonomy(SSID, MISD, SIMD MMID)

2.3.3 Universal Parallelism Rules

Process is executed by one processor with an order in serial processing. Thus, every process is executed at a single time. However, process is executed more than one processors in parallel processing. Some parts of the process can be divided to other processors to execute them, simultaneously. On the other hand, process can not be parallelise, completely. Responsible processor for identification, transmission and collection of data is called as a master processor and all of these duties are accomplished by only master processor. Identification, transmission and collection of data are accepted as serial part of the process. Thus, code, which is written to execute a process as parallel, has both serial(fraction of the code) and parallel parts.

Performance of parallel processor can be examined using Amdahl law. Fraction and number of processors are used as variables to obtain SpeedUp. Reference value is identified using fraction as 100% and number of processors as one. It is also an example of serial processing. SpeedUp, which is a function of fraction and processor number, can be calculated using Amdahl law. It is also well known and mostly applied equation by researchers. Theoretical speed up according to Amdahl is given as;

$$SpeedUp = \frac{1}{s + \frac{1-s}{N}} \tag{2.1}$$

where SpeedUp represents speed up of individual execution, s is fraction of the code and N is defined as number of processors. (Amdahl, 1967). Decrease of elapsed time for an execution can be defined as SpeedUp. Fraction of the code and SpeedUp are inversely proportional and fraction must be reduced by designer as much as architecture of process permitted . Effect of fraction of algorithm is shown in Figure 2.9.

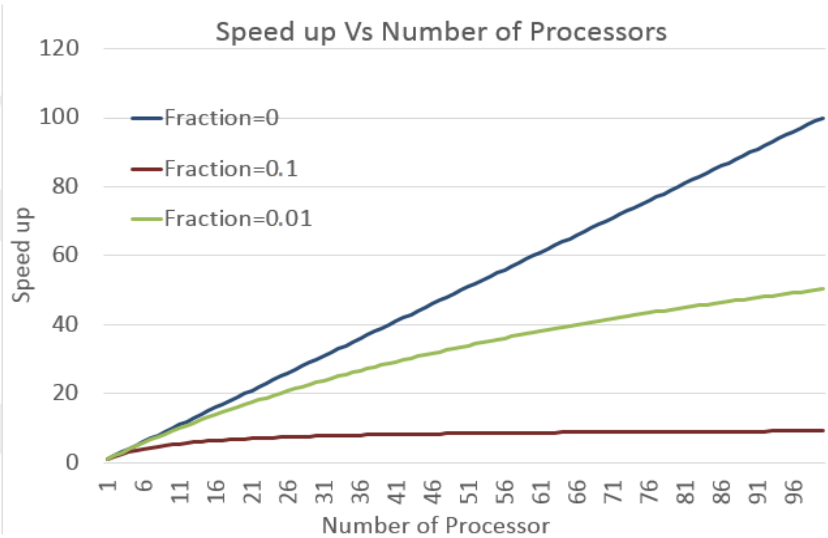


Figure 2.9 Effect of fraction of the algorithm on the speed up

Fraction of algorithm can not be predicted. Therefore, determination of fraction of algorithm is expressed by another study about Amdahl law called Karp-Flatt metric (Karp & Flatt, 1990). Fraction of the algorithm according to Karp-Flatt is given as;

$$S = \frac{\frac{1}{SpeedUp} - \frac{1}{1-N}}{\frac{1}{N}} \tag{2.2}$$

where s represents the fraction of the code, SpeedUp is speed up of individual execution and N is defined as number of processors. Speedup and fraction can be used to achieve parallel performance.

In 1993, developers, who worked in IBM, use Amdahl law as limit of parallel speed up. Code, which is written for parallel execution, can not be 100 % parallel. Thus, they claimed that, performance of the can be defined using fraction of the code (McKinney & West, 1993). In 2000, several parallel test was applied with more than one processor to show decrease of time lapse. On the other hand, speedups, which is calculated using Amdahl law, and speedups, which is achieved from the tests, were compared with each other (Velde et al., 2000).

In 2005, design of power aware system with multiple processors was tested using Amdahl law. Mathematical model was discovered for power aware system based on Amdahl law. Consumed power was also used as a parameter of the design performances. In 2009, Amdahl law is used to detect performance of hidroling modelling using Linux with C++ program (Annavaram et al., 2005).

In 2008, researchers worked in Google used Amdahl law to investigate speed up of a processors, which has more than one core. Several methods, which were used to produce processors with multiple cores, was investigated. Different equations were derived based on Amdahl law to describe methods (Hill & Marty, 2008). In 2013, researchers analyzed the effects of parallel synchronization, inter-core communication on multicore performance, speedup and scaling from Amdahl's law perspective using same formulas (Yavits et al., 2013).

In 2010, design was tested using Amdahl's law by a researcher. Direct simulation algorithm is tested over a code, which is written in the C program. Both Karp-Flatt metric and Amdahl law is used to achieve performance comparison (Nunes & Almeida, 2010). Amdahl law and Karp-Flatt metric is used to investigate the performance of the parallel execution.

2.4 Remote Monitoring

User interfaces are only way to present the measured data to users if they are not capable to fully understand the machine language. Website is designed to present the results. Thus, web-server must be installed and setted to use raspberry pi as data source. Web-site should be designed after web-server. HTML is commonly used for template of website creation. However, Database must be arranged to store and interact the measurement data. Every information must be stored carefully to used them. Interaction between database and website can not be accomplished using HTML. Therefore, PHP should be used to interact stored data and website. Web-server, web-site templates, database and interaction between database and website are the main sections of a regular website.

2.4.1 Apache

Apache is popular web-server for Raspberry Pi and it is widely used for developing web-servers. Basic programming skills is enough to install and use the program.

2.4.2 HTML

HTML is necessary to visualize the website. It is compatible for every web browser and embedded PHP programming is allowed. Graphs, tables, figures should be located using HTML. There is plenty of templates that can be found on the internet to create website freely and according to coder creativity it can be arranged to correspond necessities

2.4.3 MYSQL

MYSQL is an open-source relational database management system and it is one of the most widely used database management system. It can be installed and used in OS which belong Raspberry Pi called Raspbian. There is many tutorial how to run MYSQL over Raspbian on the internet and it can be directly accessed using the terminal. It is also highly recommended from Raspbian developers.

2.4.4 Personal Home Page (PHP)

Personal Home Page is abbreviated as PHP first but it is also known as Hypertext preprocessor now. It may be used embedded into Hypertext Markup Language (HTML) code. It can be used freely in every OS. Especially it is used MYSQL and website interactions and performance is well enough for that kind of operations. .

CHAPTER THREE

THE DESIGNED REMOTE MONITORING SYSTEM

In this part of thesis, remote monitoring of photovoltaic systems, which is shown in Figure 3.1, was accomplished. The system involves four 250 W photovoltaic panels. They are paired and each pair is connected series and each series pair is connected parallel to each other. Output current and voltage of these panels are directly connected to MPPT and battery group, which consists of four series batteries. This battery bunch is charged from the output of the MPPT.

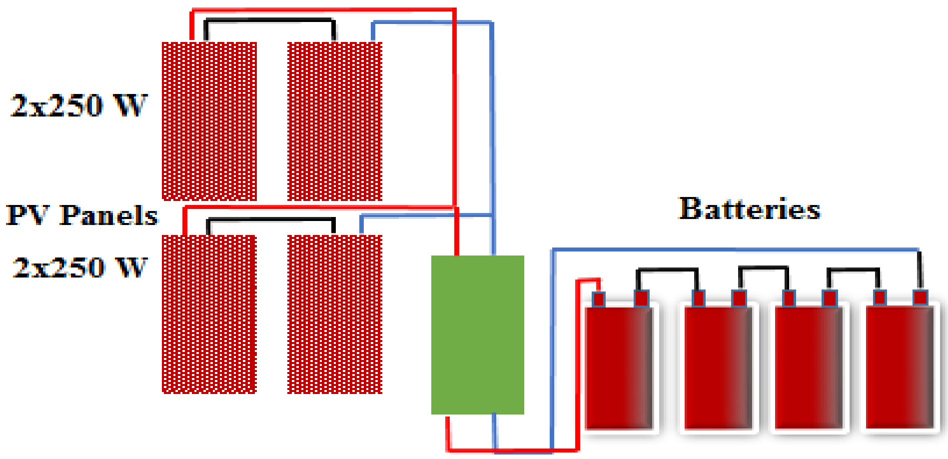


Figure 3.1 Photovoltaic plant to be monitored remotely

Block diagram of the designed remote monitoring system is shown in Figure 3.2. The remote monitoring system is responsible for processes which are conducted in the center station and remote station. Center station is the place, where users monitors the measurement taken from remote photovoltaic plant. The process is organized by a single board computer located in the center station. Measurement orders and investigation of readings are initiated over execution order of the center single board computer. The photovoltaic plant, which will be monitored on the remote station, consist of another single board computer, microcontroller and sensors. Microcontroller is responsible for converting from analog readings to digital and this information is sent using serial communication to remote station single board computer. Single board computer located in the remote station gathers the digital data and sends every readings to single board computer in the center using MPI.

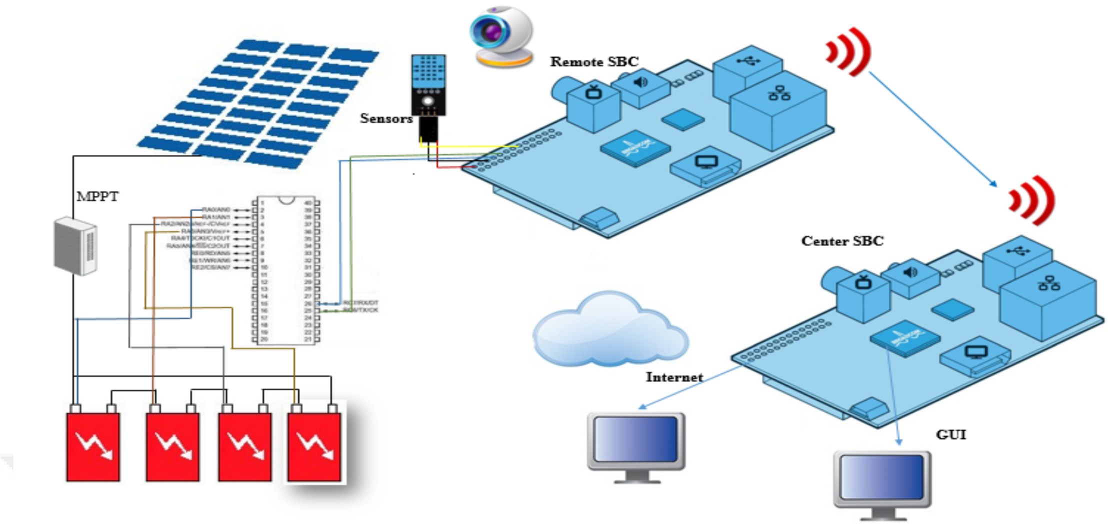


Figure 3.2 Block diagram of the designed remote monitoring system

3.1 Measurement of The Quantities in The Photovoltaic Plant

To monitor the photovoltaic plant remotely, some quantities need to be measured or calculated. Measured quantities are input-output voltage- current of MPPT, voltage of each battery, temperature, irradiation, dust density, humidity and gain of charge controller. Sensors, which are used to sense some of these quantities, give digital information. The outputs of these sensors are directly connected to the Raspberry Pi located at the remote station. For some other quantities, analog information needs to be converted into digital form. This task was introduced using PIC-18F4580 microcontroller (Köklü & Kılınç, 2016).

Any kind of single board computer can be used on the designed remote monitoring system to monitor the measured quantities because their design purposes are based on same principles. However, GPIO pins, which are used to read measurements, and predefined libraries are the key factor to chose Raspberry pi. Microcontrollers require additional settings, codes and circuits to operate with displays. Another displays, such as tft-lcd, can be used after manipulation of the related display’s driver file.

3.1.1 Temperature and Humidity Measurement

Temperature is an important parameter for batteries. They can be used or charged under 45 Celsius to run them in an optimum condition. Regular readings of humidity and temperature protect devices from aging (Al-Dahoud et al., 2014). DHT-11 sensors was used to measure temperature and humidity. Reliability and stability tests of the sensor were performed in order to ensure about the results. Another reason of choosing the sensor is connection diagram, which is pretty basic and simple (Figure 3.3). Three pins, which are GND, VCC and Data, are required to complete the connection. Sixth pin of Raspberry pi was selected as a ground and supply voltage (VCC) of 3.3 V was applied using first pin. Another pin is needed to transmit data from the sensor to Raspberry pi. Detailed analysis about connections of pins to sensors prevents from complication of circuits. The pin can be chosen by user regarding to another existing connections. However, chosen pin must be declared in the code, which is written to read from output of the sensor.

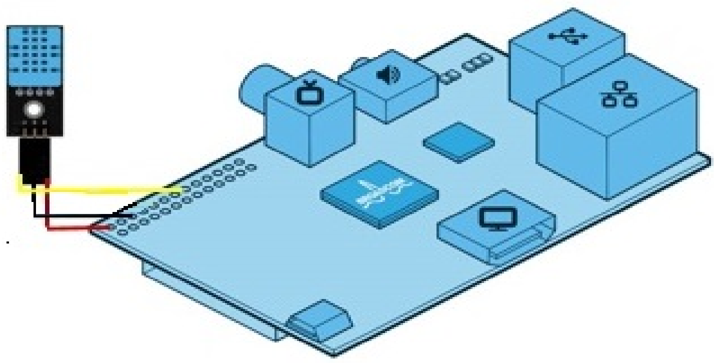


Figure 3.3 Connection of the DHT11 sensor to the Raspberry pi

Python program was used to read data because libraries, which are needed for almost every process on any programming language, can be found as an open source. Import operation is initially needed to use features of libraries. Reading pin and sensor type must be declared in the program. Reading sequence must be written after initialization. Basic code to read temperature and humidity information can be seen in Figure 3.4.

This code can not be used directly to read temperature and humidity information in larger operations, which involve multiple sensors. Outputs of this code are printed temperature and humidity informations. However, algorithm blocks the other functions because it is not designed for other purposes. To avoid from blockage threating of other functions, which allows to use functions simultaneously, or another algorithm without any changes in core of the code should be used. Both threading and another algorithm are used in the study for different purposes. Threading is used to check communication with sensor and different algorithm is used to read the regular measurement.

```
import Adafruit_DHT
sensor = Adafruit_DHT.DHT11
pin = '4'
humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)
while True:
    if humidity is not None and temperature is not None:
        print 'Temp={0:0.1f}*C  Humidity={1:0.1f}%'.format(temperature, humidity)
    else:
        print 'Failed to get reading. Try again!'
```

Figure 3.4 Concept code for temperature measurement written in Python

3.1.2 Dust and Voltage Measurement

Dust measurement is accomplished using Sharps's GP2Y1010AU0F sensor and voltage readings of each battery were completed using voltage divider circuit. These two measurements must be converted to digital data before sending them to center station because the output of both operations are analog. As a converter PIC-18F4550, which has 11 analog inputs, was used.

PIC should be programmed over USB port of a computer with a programming circuit and there is plenty of program, such as PIC basic and CCS C, to write the functions, which will run on PIC. Therefore, language level of many programs are lower than CCS C. Thus, simple functions, such as floating operations, are not predefined processes in most of them. CCS C can be coded like C and this kind of features are the main reasons to choose CCS C for programming purpose.

Analog to digital conversion is completed on PIC and digital information is sent over serial channel. Timing of data writing and reading is another crucial issue to avoid from missing information during the process. Both PIC and Raspberry Pi must be initialized by considering in same time delay to work synchronously.

As mentioned in beginning Sharps's GP2Y1010AU0F sensor was used to sense dust particles around the photovoltaic panels. In this sensor, a diode and a transistor were located to sense dust in the region. Performance of the sensor was considered highly acceptable since it is commonly used for air purifier systems. 20-11 mA current consumption was acquired. Analog output of the device is directly proportional to the measured dust density and sensitivity of the sensor is 0.5V-0.1mg/m. Sensor and written code can be seen Figure 3.5.

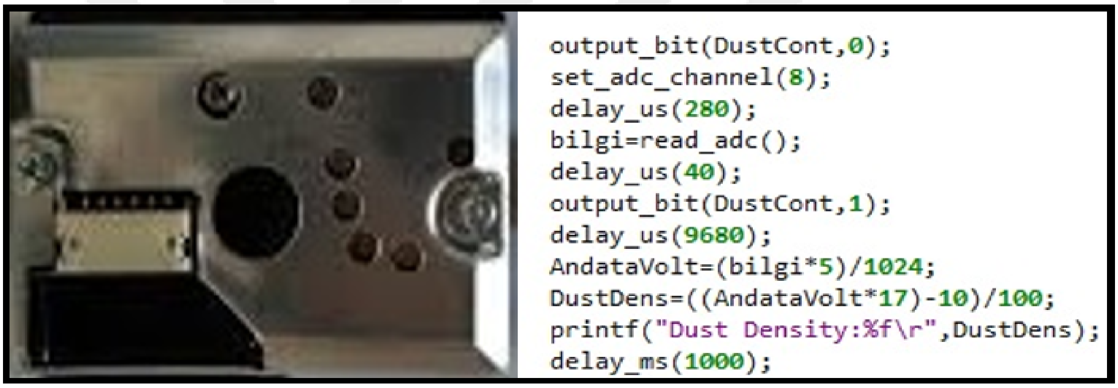


Figure 3.5 Dust sensor and concept code to read data from the sensor

Voltage measurements were taken using voltage divider principle. Working principle can be basically explained that potential difference is measured over a resistance from the circuit consisting two resistance. The main reason for using voltage divider in the study is voltage reference of PIC is defined as 5 V. Thus, maximum value, which PIC can read, is 5 V. Maximum and minimum voltages should be re-defined by proportionally changing it between 0-5 V in order to correctly read the voltages.

On the other hand, information, which is read on the related pin, is transformed to it's real value using resistances value in the code. However power dissipation on resistances must be under maximum power of resistance and it is typically 0.25 W.

Trimpots are commonly used for dividers and they are also preferred in this study.Input and output voltages of charge controller and voltage of four batteries are measured. Simulation was done using Proteus program and the monitored data in the simulations of the voltage readings can be seen in Figure 3.6.

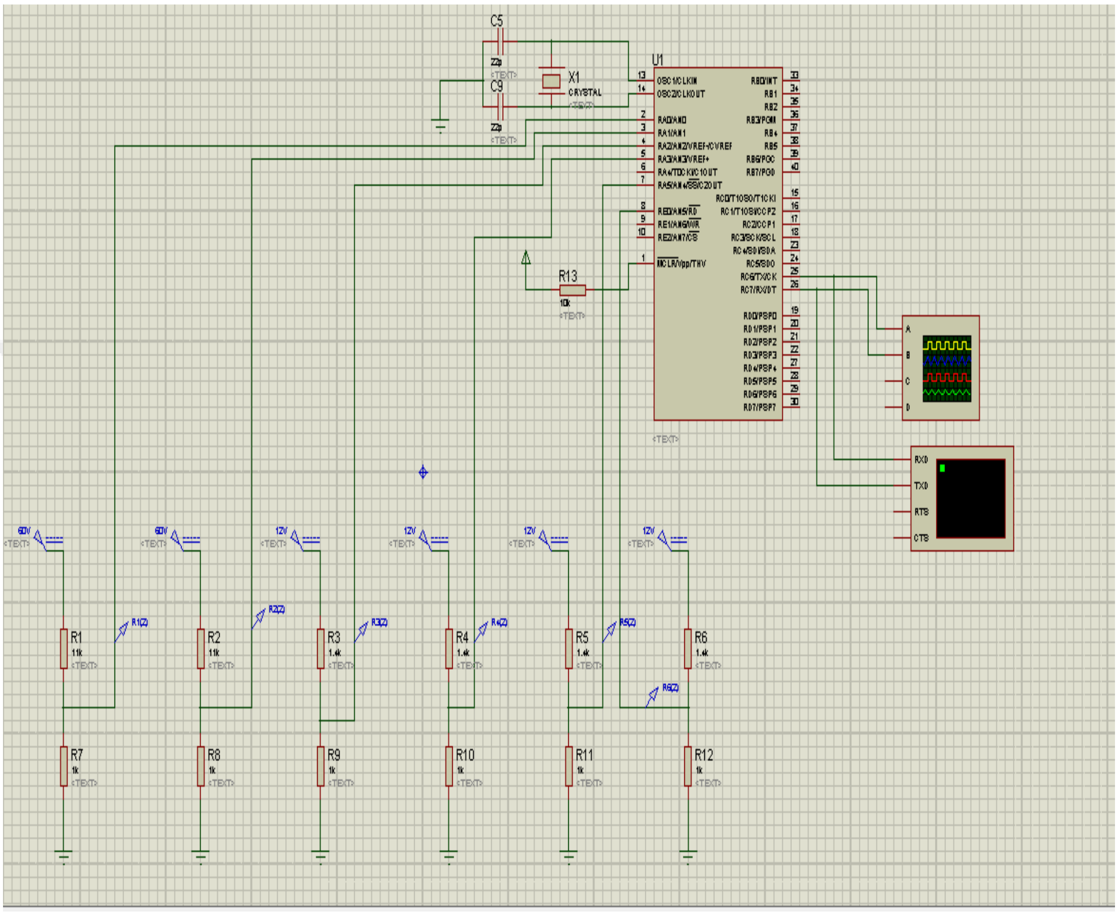


Figure 3.6 Simulation of voltage readings of batteries

Dust density and voltage measurements were completed using voltage dividers and dust sensor circuits as mentioned before. A design of circuit for these two measurement is completed and the circuit can be seen in Figure 3.7.

All measurements were performed using PIC and using Raspberry Pi. Voltage measurements were taken by voltage dividers. Dust densities were taken using Sharps's GP2Y1010AU0F sensor and output data of the sensor were transformed to desired unit. All these analog values were sent over serial channel to Raspberry Pi. All these data are sent over MPI to center Raspberry Pi.

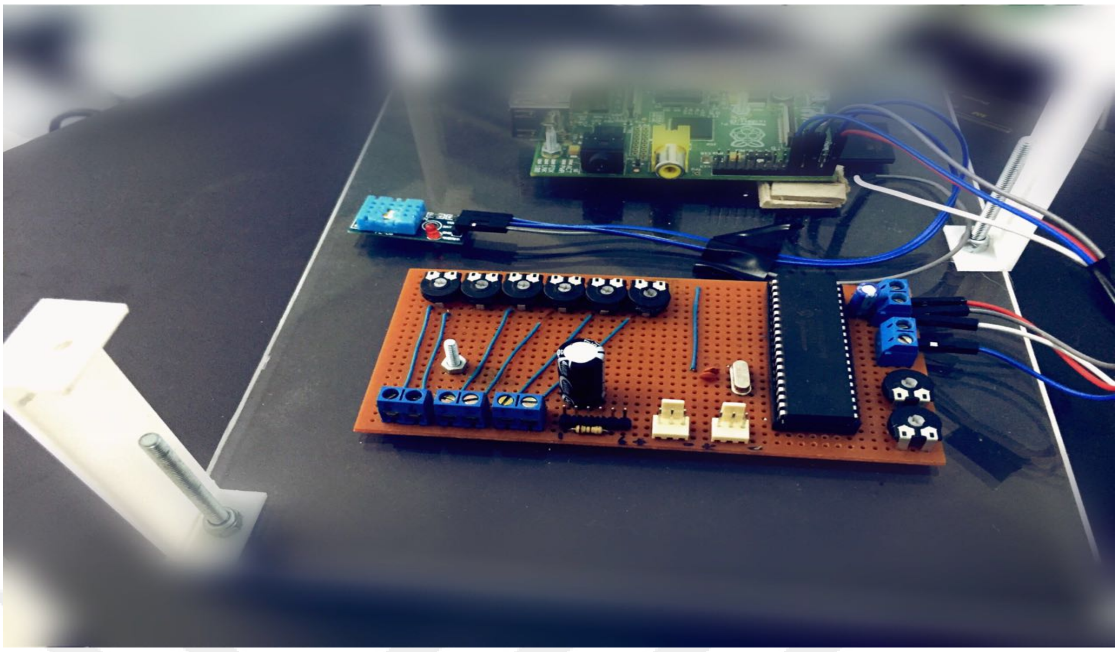


Figure 3.7 Circuit for converting analog dust and voltage information to digital

3.1.3 Current Measurement

Current measurement of photovoltaic system is taken using non-invasive method in both side of MPPT. Current measuring device is not connected series to circuit in non-invasive method. Thus, current sensor can be clamped around the supply line. Hall effect measuring principle is used in this kind of sensors. Has-50s is used as a sensor and it can measure current up to 50 amp with galvanic isolation. Current sensing method and sensor can be seen in Figure 3.8.

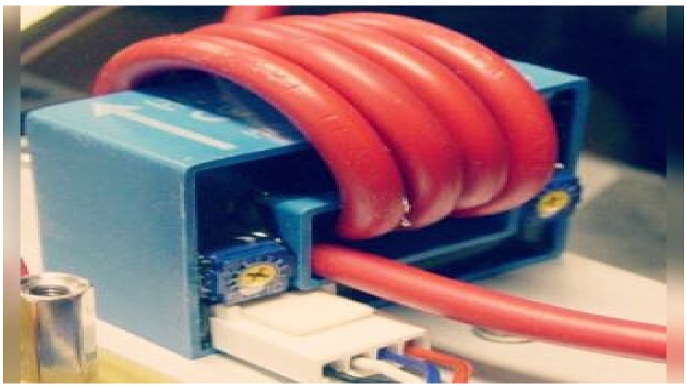


Figure 3.8 Current sensing method and sensor

Output of the sensor is changing between -4 V and +4 V under reference direction of the sensor. Current sensor output is directly connected to PIC because output of the sensor is analog. However, 0.19 mV alteration can be notice using 8 bit analog readings. Thus, 0.25 mA resolution is defined based on this scope. On the other hand, additional supply circuit was designed to use the sensor because supply voltage of the sensor is +15 and -15. In the remote station, there is no source in this voltage level so 1 transformer, which transform 220 V- 15 V, and 7815 and 7915 regulators are used in the circuit. Circuit simulation can be seen in Figure 3.9.

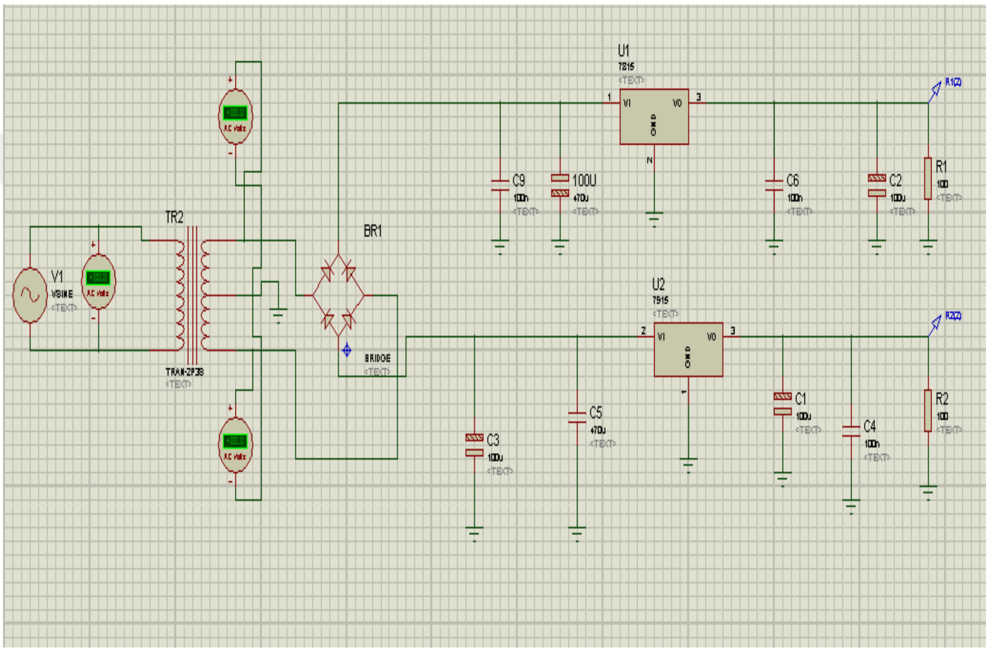


Figure 3.9 Supply circuit of current sensor simulation

Simulation and circuit has some differences because of the inner resistance of 7815-7915. Output of the negative and positive side must not be exactly -15 V or 15 V. However, supply specification of the sensor is restricted with -14.75 to -15.25 for negative pin or +14.75 to 15.25 for positive pin. Thus, output resistance of the circuit is setted using potentiometer. Circuit can be seen in Figure 3.10.

Output of the MPPT directly connected to the battery group. Hence, battery group is load of the system. However, each battery, which are connected to the photovoltaic plant, is fully charged before current measurement. Therefore, current, which is measured in the output of the MPPT, is zero. A spindle was connected to the output of

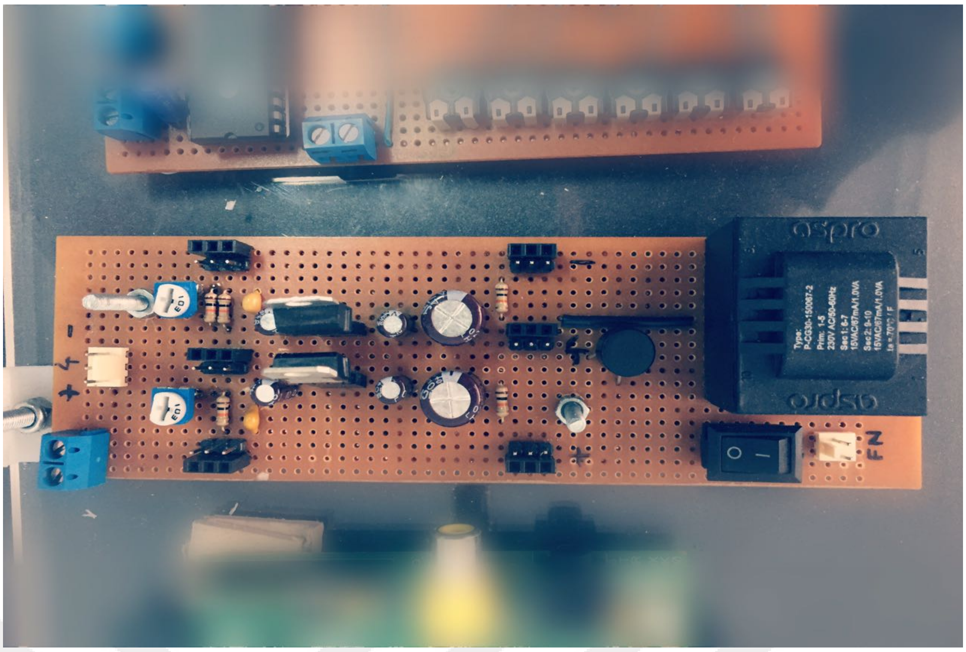


Figure 3.10 Current sensor supply circuit

the MPPT to investigate current measurement. Block diagram of current measurement can be seen in Figure 3.11.

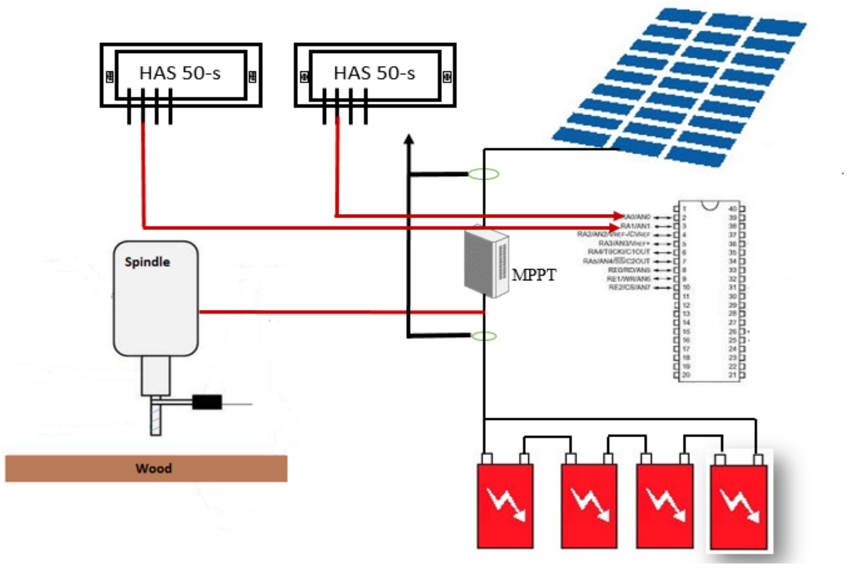


Figure 3.11 Block diagram of current measurement

3.1.4 Irradiation Measurement

Irradiation of the photovoltaic panels can be measured using sensors or short circuit current of extra photovoltaic panel. Short circuit current of an extra photovoltaic panel is used to read solar irradiation. Specification of the photovoltaic panel can be seen in Table (3.1)

Table 3.1 Technical specifications of photovoltaic panel

Technical Specifications	
Peak Power	10 W
Peak Current	0.55 A
Peak Voltage	18.18 V
Short Circuit Current	0.57 A
Standard Test Condition	1000 W/m^2 25 °C

Short circuit current of photovoltaic panel and irradiation of it can be considered directly proportional with each other. Location and angle of the reference cell is extremely important to obtain required quantity (Figure 3.12).

Figure 3.12 Irradiation sensor assembly in remote system

3.2 Processor’s Communication Methods

Communication method used in the study was determined considering the region where the solar system was established. In this thesis, remote monitoring of photovoltaic system was designed for low powered plants near urban areas. Internet connection within EDGE, 3G etc. can be easily accessed for systems located urban areas. Same network has been already distributed to the universities or factories. Communication can be done using the internet connection without any extra cost. Speed is enough for transmit the measurement data.

Communication using internet can be done using TCP/IP. MPI program, which was designed for parallel communication, was considered as one of the most suitable one for our study since measurement orders can be simultaneously executed and examination of the data can be performed under parallel processing procedure and protocol. Data examination was applied over a processor in a common way but examination of parallel processing data was divided to remote stations and sent over MPI. Both serial and MPI based communications were completed. These are serial communication between microcontroller and Raspberry Pi in the remote stations and the communication between remote Raspberry Pi and center Raspberry Pi.

3.2.1 Serial Communication Between Raspberry Pi and PIC

Common protocol is necessary to swap desired information among processors. Suggested protocols and methods can be classified under two major groups which are parallel and serial. Serial communication can be basically expressed as transmit or receive one bit at a single time. SPI, I2C, UART are well known serial interfaces. The data flow of serial communication is shown in the right side of the Figure 3.13.

Asynchronous and synchronous methods are the parts of serial communication. External clock pin is required to initiate synchronous for timing purpose. Asynchronous method is an acceptable one for minimization of using pin whereas data traffic should be carefully controlled to check reliability of the results. UART is a circuitry that is responsible for carrying out serial communication. Wiring can be seen in the left side of Figure 3.13.

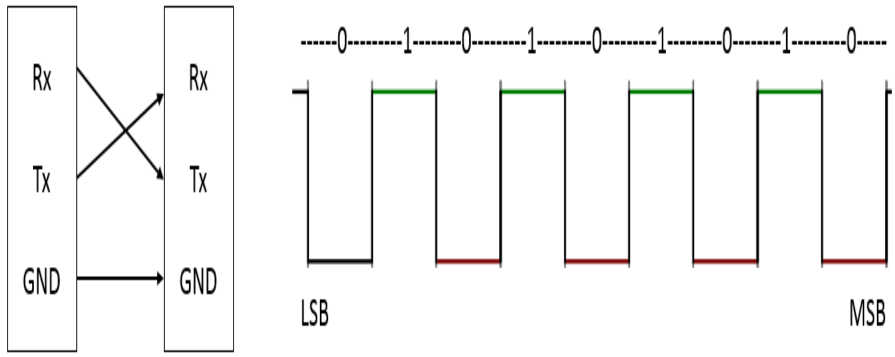


Figure 3.13 Serial communication wiring and data flow

Measurement data should be sent over serial channel using UART pin, which were embedded in both Raspberry Pi and PIC. At the beginning, line can be found in /etc/inittab folder and it must be commented.

comment the line "T0:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100" .

After that, a line,which can be found in boot folder and cmdline.txt file must be removed to avoid data sending from Raspberry Pi to PIC. The line is shown in bottom.

“console=ttyAMA0,115200 kgdboc=ttyAMA0,115200”

After the initialization was performed by a program, which is called minicom, data flow can be tested as shown in Figure 3.14. Minicom is a program, which can be used to observe data flow on serial channel.

There is an extra issue in serial communication, which is known as voltage level differences. PIC was supplied with 5 V and Raspberry Pi works with 3.3 V in serial channel with reference used in this study. Therefore, there must be voltage divider between the serial pins of the processors. Connection diagram can be seen in Figure 3.15.

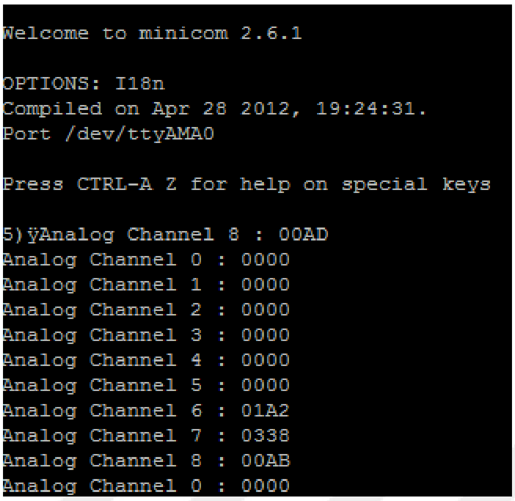


Figure 3.14 Test of serial communication using minicom

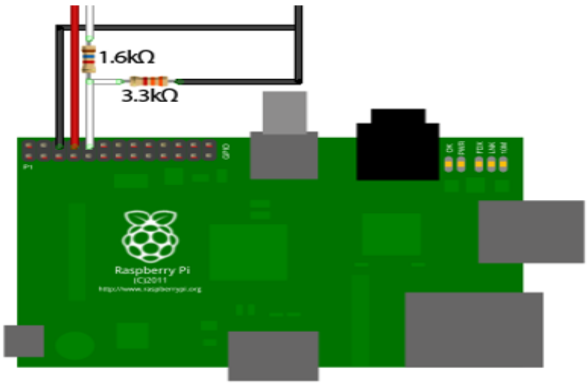


Figure 3.15 Designed voltage divider for serial communication

In programming, baud rate, which is speed rate of the communication, must be identified. There is plenty of alternatives, such as 4800 and 9600, about baud rate. However, it must be identified in same range for both Raspberry pi and PIC to read data. Before expression of the baud rate, library must be imported. After these operations, serial port must be opened and data flow should be checked.

A code, which is designed for listening the serial channel, can be seen on Figure 3.16. However, it can be used to listen serial channel only. There should be timing or threading to simultaneously notice all measurements.

```
import serial
ser = serial.Serial('/dev/ttyAMA0',9600,timeout=0.8)
ser.open()
dosya = open ("/home/pi/Desktop/deneme.txt","w")
        while True:
                for line in ser.read(1):
                        try:
                                response = ser.readline()
                                print line + response
                                dosya = open ("/home/pi/Desktop/deneme.txt","a")
                                dosya.write(line + response)
                        except:
                                ser.close()
```

Figure 3.16 Written code in Python for serial communication

3.2.2 Communication Between Remote Raspberry Pi and Center Raspberry Pi

Communication between center and remote Raspberry Pi was accomplished over MPI. MPI was designed for parallel processing on academia. It can be used freely, it is not for commercial usage. Most of the MPI distribution can be used directly in C, C++, Fortran. However, free libraries can be found for program, such as Python.

There is a master/slave relation between parallel processors. Master processor manages the running operation. Master can be use for data distributor and collector. If master processor is also used in operations, Speed up, which is expressed as a decrements of time lapse during the operations, will be higher. Master processor is used as a collector in communication parts of the thesis.

Algorithm, which is written in Python, of MPI includes three main parts. First part of the algorithm is about start order. Start order is sent from master processors to remote stations. Measurements are taken by processor in remote station and second part of the algorithm is created with this operation. Measured information is sent to center station from remote station in the last part of the algorithm.

Furthermore, MPI is also used for parallel processing in this design. Algorithm, which is written in Python, of parallel processing consists of three part. First part of the algorithm is about data distribution and it is directly related to master processor. Thus, related data is distributed over master processor under equal work load according to available processors count. Every related data are executed in their own processor and this operations create the second part of the algorithm. The last part of the algorithm is collection and monitoring of the data in master processor. Information, which is result of operations, is collected in master processor and then monitored using available hardware.

Examination process is done by more than one processor and it takes less time. Embedded computer cluster has been created using this method and it should be tested to prove optimization for time lapse. All single board computers used in the design are benchmarked by running the Linpack algorithm, in single precision mode.(Figure 3.17). Matrix linear equation is solved to assess the used single board computer ability to solve real world problems (Cox et al., 2014). A range of different LINPACK problem sizes are investigated using the results of single precision benchmark. Figure 3.17 illustrates the results, which indicate a peak single node performance of around 69400 kflop/s for a problem size n = 120.

The results presented in figure 3.18. Figure 3.18 is showing the performance of designed cluster when it is running with HPL This benchmark solves a dense n x n system of linear equations, Ax = b, various problem sizes n and numbers of nodes N are used in benchmark. The results are plotted as a function of the problem size, n, for each of the numbers N of nodes in Figure 3.18. It is not possible to run the 1024 problem size on only one node because the available memory was insufficient.

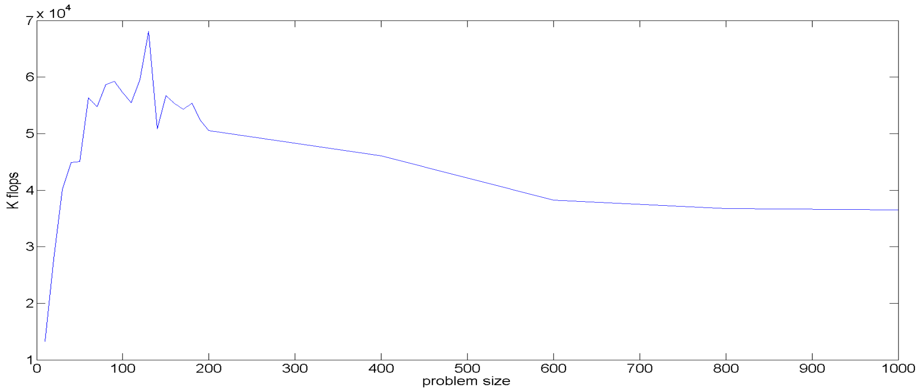


Figure 3.17 LINPACK benchmark performance of a single node on various problem sizes in single precision mode (test applied on Raspbian for every processors in the design).

Performance scaling becomes almost constant (problem size=600) when the problem size is sufficiently high. The performance is presented in Figure 3.18.

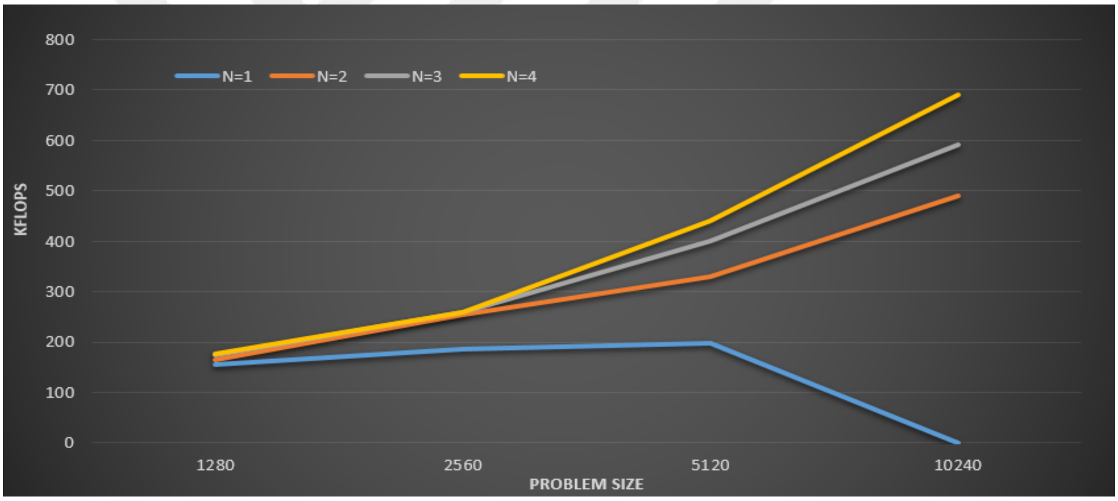


Figure 3.18 Performance of HPL benchmark, as a function of the various order n of the matrix A, for various numbers N of processor number(test applied on Raspbian for every processors in the design).

Communication for execution orders, data transmission and examination of data was done using MPI. Processes can be done faster and communication is accepted very robust using MPI. MPI was coded over Python but syntax of MPI is slightly different. Same code can also be executed in every station for a method, which can be used to compute different data pool using same operations. However, different code can be executed in every node if it is saved under the same name and directory. Different code execution method is useful to solve one problem with different methods using same data pool and same code execution one is useful to decrease time lapse.

3.3 User Interfaces

A User interface was designed to observe and control the measurements. Graphics, tables and colorful objects were used to create effective illustration for the user. Website and GUI were created as a source of user interaction. Website was designed to visualize the data under login protection. Graphical user interface was designed to visualize data, report the measurements and test the designed system.

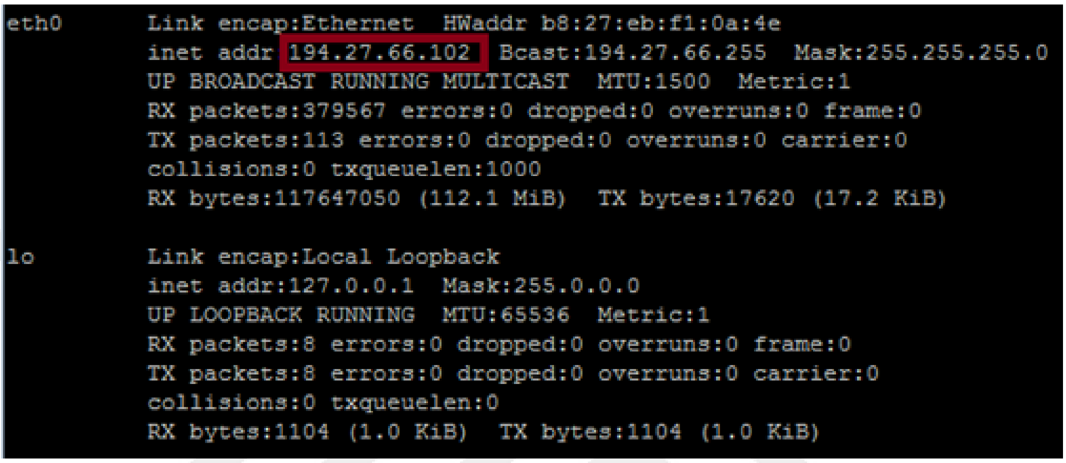
3.3.1 Website

Website contains information about project, methods and codes used in the project. Additionally, graphs and informations about measurements can be observed using website. Every code and tried method, which involves falsified and verified ones, were shared and they were accepted as an open source techniques. Every code and method is free to use and open source. Study can be developed by everyone without any hesitation.

Web-server is essential necessity to build a website. Hyper Text Transfer Protocol (HTTP) call is created over web-server and it is generated to publish data on the world wide web (www). Storage of the information and delivery the information to client were fulfilled by web-server. Apache, IIS, nginx and GWS are widely preferred web-server programs. Apache web-server program was chosen to reach the purpose, because it is popularly used web-server program and versions of it can be easily found for every OS freely.

3.3.1.1 *Website Design and Creation Methods*

Apache is popular web-server for Raspberry Pi and it is widely used program. Basic Linux programming skills is enough to install and use it for the web-server. It works in the background. First of all using ifconfig command ip address can be seen on terminal as seen Figure 3.19 and it should be noted.



```
eth0      Link encap:Ethernet  HWaddr b8:27:eb:f1:0a:4e
          inet addr 194.27.66.102  Bcast:194.27.66.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:379567 errors:0 dropped:0 overruns:0 frame:0
          TX packets:113 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:117647050 (112.1 MiB)  TX bytes:17620 (17.2 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1104 (1.0 KiB)  TX bytes:1104 (1.0 KiB)
```

Figure 3.19 Method for identification of IP address

After that Apache can be easily installed on the machine terminal using simple method like;

sudo apt-get install apache2 -y

It can be checked using the IP address. The result should be like Figure 3.20.

After installation of Apache index.html file can be investigated in /var/www folder. Index.html file is the main file of the website. Other files in the folder can be used by referencing with index.html file. PHP or HTML can be chosen as other file types. In this study three pages including main page exist. They have all used embedded PHP to interact with MYSQL.

MYSQL was used for data storage in the study as database and it can be installed using;

sudo apt-get install mysql-server

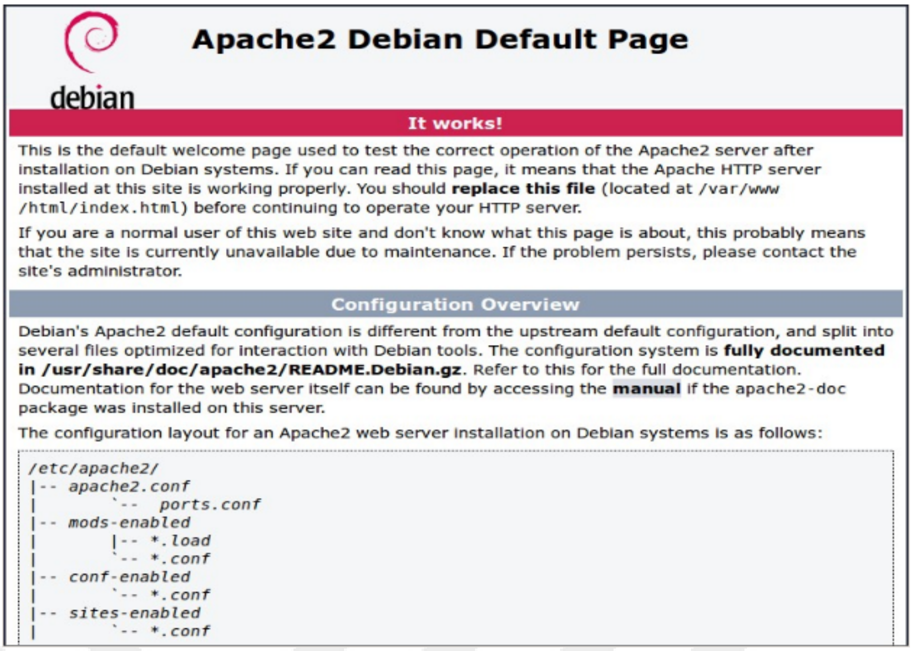
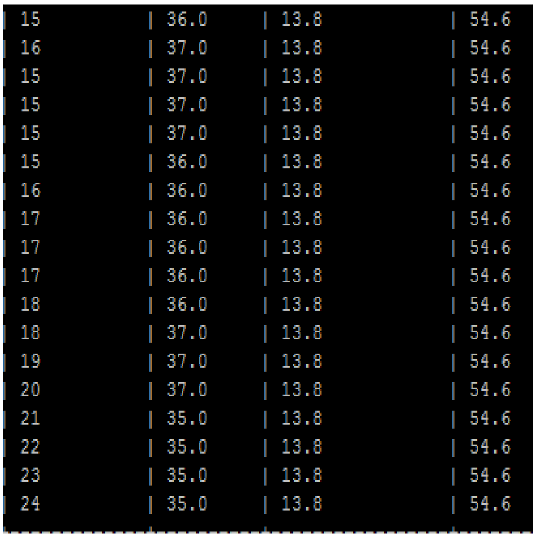


Figure 3.20 Test of web-server over local website

After installation of the program, databases can be created over terminal. Tables can be created according to remote stations. Every remote station has its own table to store measurements. The measurements were composed of voltage of each battery, dust density, temperature and humidity and infinite rows were identified in initialization of every tables. Table, which was created to store measurement data, can be seen in Figure 3.21.



15	36.0	13.8	54.6
16	37.0	13.8	54.6
15	37.0	13.8	54.6
15	37.0	13.8	54.6
15	37.0	13.8	54.6
15	36.0	13.8	54.6
16	36.0	13.8	54.6
17	36.0	13.8	54.6
17	36.0	13.8	54.6
17	36.0	13.8	54.6
18	36.0	13.8	54.6
18	37.0	13.8	54.6
19	37.0	13.8	54.6
20	37.0	13.8	54.6
21	35.0	13.8	54.6
22	35.0	13.8	54.6
23	35.0	13.8	54.6
24	35.0	13.8	54.6

Figure 3.21 MYSQL measurement table, which consist all measurement data

Name, surname and duty columns were created of login table and these tables can be directly modified by the managers using website. However, managers and users are restricted with direct login. If their login information matches with the managers, registration information can be erased or generated by them.

In the study, PHP was used for database interaction and visualized data was taken as related table over PHP embedded HTML code. Embedded PHP was also used in Python code.

PHP can be installed easily on Raspbian, which is an OS created for Raspbbery pi, like;

sudo apt-get install php5 libapache2-mod-php5 -y

After that, in the /var/www folder it can be used for database interaction. Website of the study was designed using HTML. There are two pages which are accessible for users, who do not have username and password. Main page involves general information about project and it can be seen in Figure 3.22.

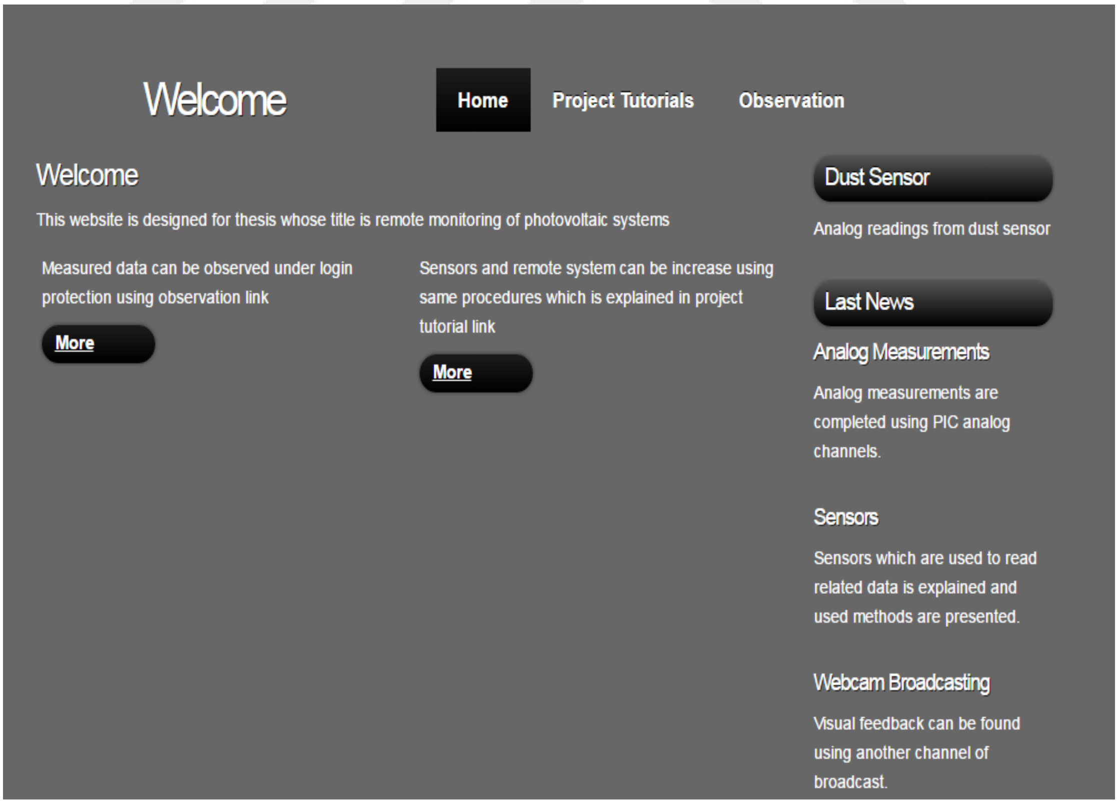


Figure 3.22 Homepage of the website designed for the remote monitoring system

3.3.1.2 Features

Website is not designed totally under login protection because methods can be investigated and copied freely to share information in an open source way. Measurements and user information can be visualized under login protection and two login group as users and managers were identified in the scenario. Only measurement data can be seen by the users. Login page of the website can be seen on left side of Figure 3.23. Authorized operations for managers can be seen on right side of the Figure 3.23.

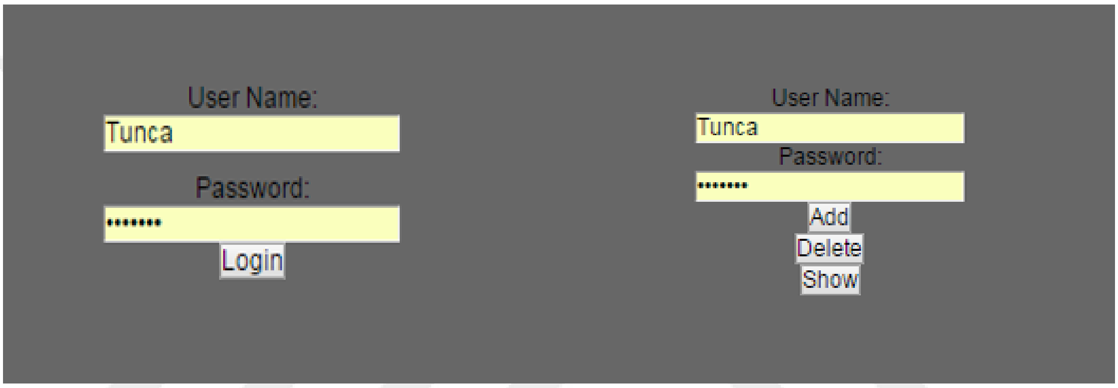


Figure 3.23 Login and authorized operations for managers of the website

All verified methods, codes and circuits can be found under Project Tutorials title. Used sensors and their circuits, written codes, libraries and how to install them were explained step by step to give introduction to users. Tutorial page of the website is shown in Figure 3.24.

3.3.2 Graphical User Interface

Graphical user interface of the study was designed using seven inch display. It was not originally designed to use on single board computer but calibration and modification in driver files were prepared to run it over single board computer. Every display should be calibrated regarding to single board computer resolution using its own driver. HDMI - VGA driver board was used to interact between Raspberry Pi and the display. Low voltage data signal (LVDS) output was directly connected to the

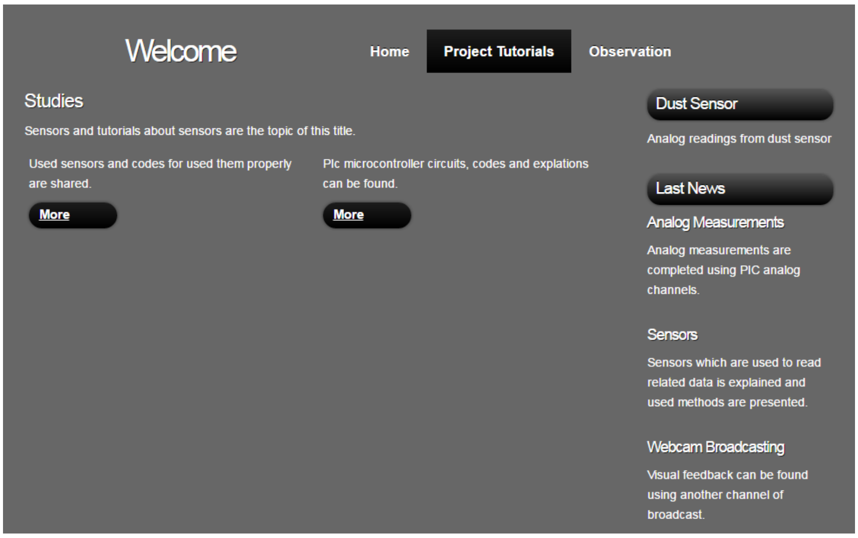


Figure 3.24 Tutorial page of designed website seen on seven inch TFT LCD display

driver board and HDMI output of the driver board was connected to Raspberry Pi.

Tkinter is used for programming the GUI and it is most commonly used as a library for GUI. Main page of the GUI can be seen in Figure 3.25. Button group was designed to monitor and test the measurement in top-center on the main page.

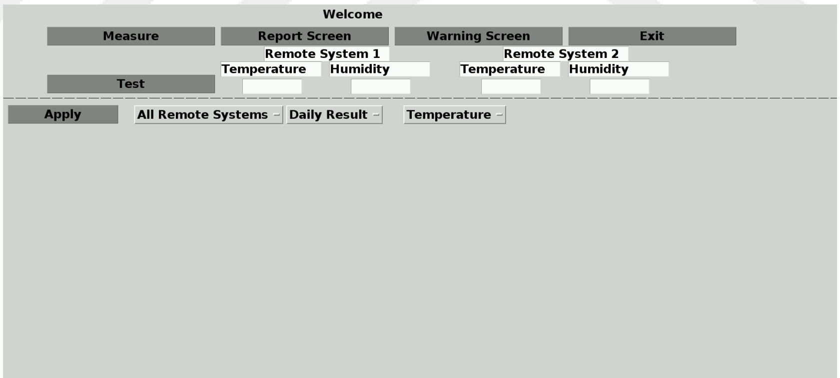


Figure 3.25 Main page of the graphical user interface seen on seven inch display

Measurements of the data are begun by pressing the Measure button and other buttons can be used when the Measure button was pressed. Test button was created to ensure communication between center and remote stations. When this button is pressed, remote station's temperature and humidity data are visualized on the display. Exit button serves to close GUI to explore Raspberry Pi desktop.

A button and three option menu are located in the center of the GUI. Button, which is called Apply, is sent the drawing order based on option menu results. Default values were identified to avoid empty results error. First, option menu was set to be responsible for choice of remote systems. Remote system 1, remote system 2 and others (all remote systems) can be selected from the option menu. Title of the database table was taken using this results. Time interval information is identified as an output of the second option menu and daily or monthly results can be selected from option menu.

Downloadable content can be set using the first and third option menu and download order was sent using the second option menu. Titles of the graph are the result of third option menu and voltage of each battery, temperature, humidity and dust can be selected from the option menu. When the user presses the apply button, the graphs are automatically drawn. However, if remote system 1 and remote system 2 is not selected together, only one graph is able to be visualized on the screen. All remote system data visualized under graphs when all remote systems are selected together.

Report screen was designed as a totally different screen. It was opened when Report Screen button pressed and it can be destroyed using Exit button located in main page. Five visualization tools were created to organize report operations. Three option menu was coded for the same reason in the main page of GUI. Their outputs were the same but outcomes of the buttons were visualized the data with text views. First page of the report screen can be seen in Figure 3.26.

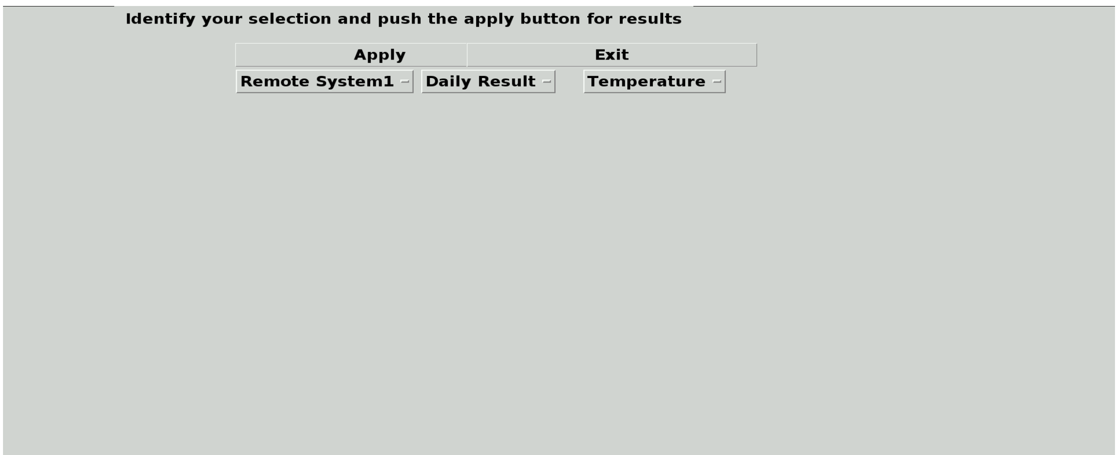


Figure 3.26 Main page of report screen seen on seven inch display

Examination of unexpected data was identified in the warning screen. Parallel processing principle was used to investigate the data. Three button, one edit text and one text view were located on the screen. Exit button was responsible for canceling the related screen. Processed data were specified under output of edit text or option menu. Virtual keyboard was visualized when edit text was triggered. Buttons, which are defined on a frame, was used for creation of keyboard. Apply section was kept as the starting point of operation after every essential input was entered. Database connection over Python was used the essential inputs for processing the desired data. Main page of the warning screen can be seen in Figure 3.27.

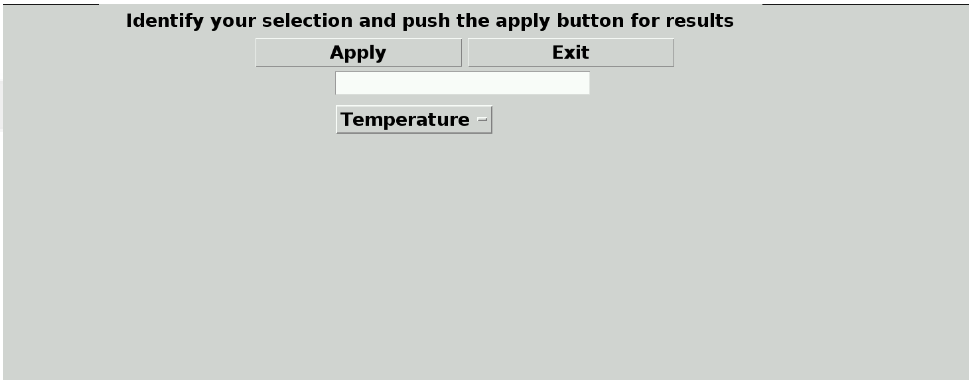


Figure 3.27 Main page of the warning screen seen on seven inch display

Information is controlled using limits and informations under or over the limits sent back to the center station to observe them. Limits are identified using datasheets, specifications of related devices and measurement information. Maximum temperature of the batteries is defined as 45 Celcius. Battery charge operation is observed and minimum and maximum voltage value for each batteries is identified using measurements. Each battery is fully charged so the voltage readings are always same. Output of the MPPT is connected to a load to measure current in both side of the MPPT. Current consumption rate of the load is also considered and they are used as limitations of currents. Current consumption of the load is changing between 1.3-2.7 A.

On the other hand, output voltage of the MPPT can be known voltage requirements of the load too. It is also identified in instruction of the plant. Output voltage of MPPT is 55 Volt. Thus, these limits should be changed based on established system. All these designs, codes and used components can be seen on website and can be used by related users. They are all open source design.

3.4 Implementation

Boxes was prepared to use the system properly. Center Raspberry Pi box is designed as three part and the first part of the center Raspberry Pi box was power unit. Two switched mode power supplies (SMPS) were used to correspond to the power requirement. First SMPS was AC-DC step down switching regulator which was transformed the 220 V to 12 V. Supplied current rate was 10 A an it is enough to correspond the requirements. Second DC-DC converter, which transforms the 12 V to 5 V, was used to power Raspberry Pi and display. Both of the devices work with 5 V and Raspberry Pi draws current of 700-900 mA.

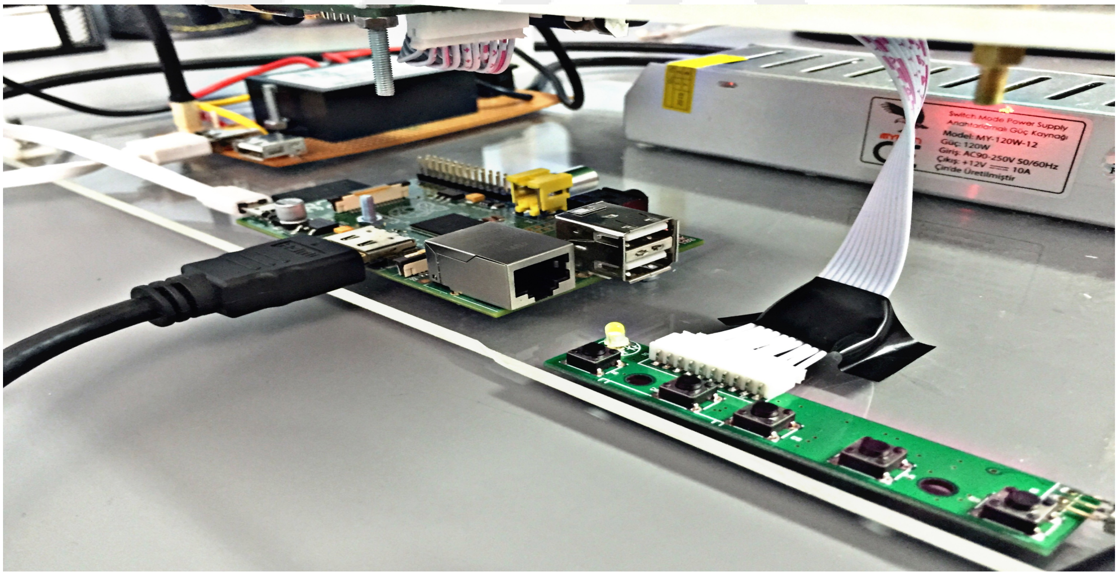


Figure 3.28 Inside of the center Raspberry Pi box

Second part includes monitor and monitor related hardwares, such as drivers and button groups. Seven inch display was used and its driver is VS-TY2662-V1 circuit, which can directly connect using HDMI and VGA outputs. Power input should be connected over driver. Button group was used to change input type such as HDMI to VGA, display settings and on/off conditions. On the other hand, the last block involves

Raspberry Pi and its connection. Box was illustrated in Figure 3.28 and Figure 3.29 .

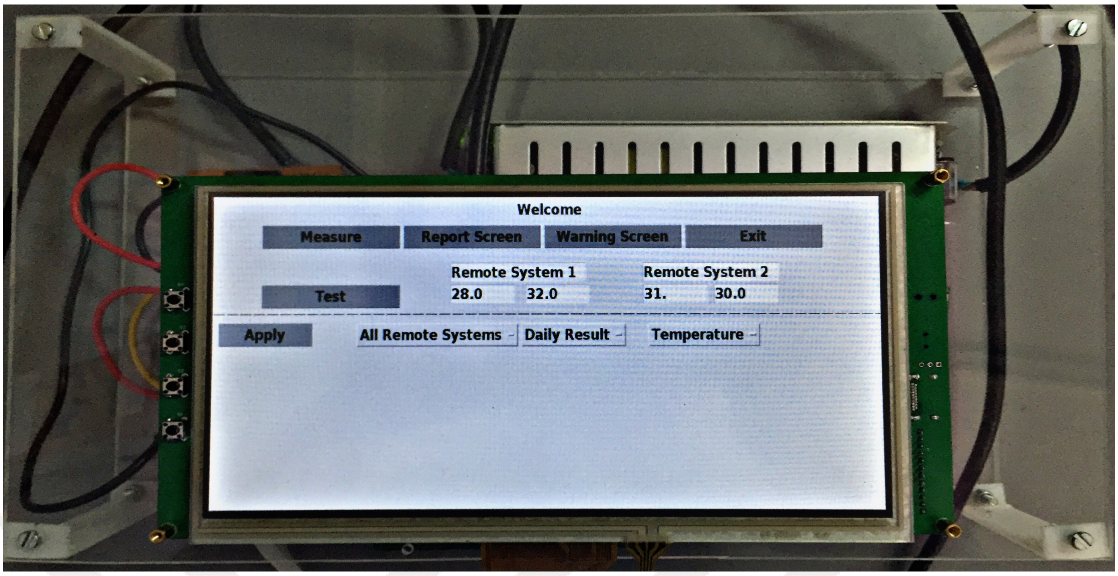


Figure 3.29 Outside of the center Raspberry Pi box

Another box was designed for remote stations. Remote station was powered using power banks and it was designed as two parts such as circuit board and Raspberry Pi. Circuit of the remote station is responsible for measurement and their transmission to Raspberry Pi. It can be seen in Figure 3.30.

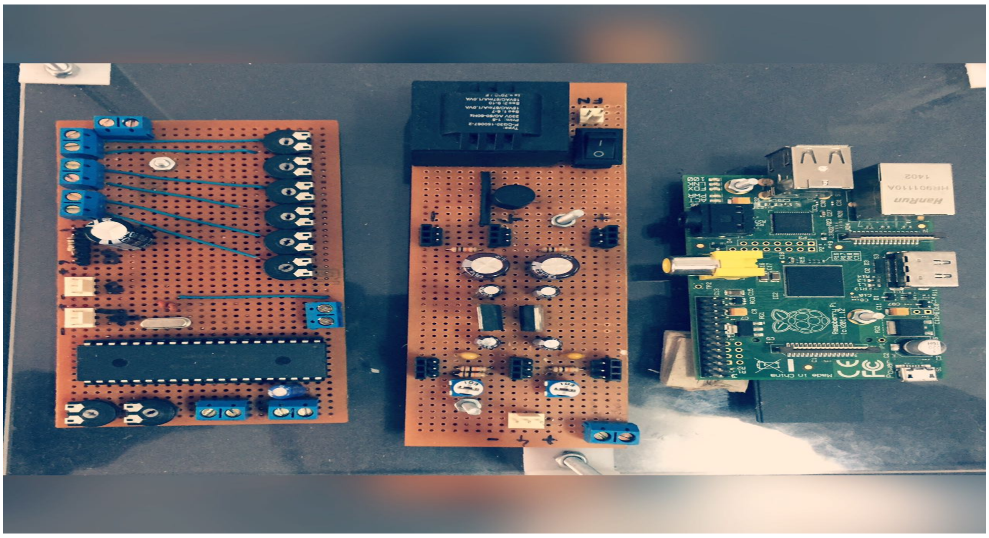


Figure 3.30 Box of the remote stations including circuit and Raspberry pi

CHAPTER FOUR
RESULTS

The results were mostly presented by visualizing the measurements in order to clearly show the importance parallel processing performance. Measurements are visualized using GUI in the related screen and website. Parallel performance of the system was presented to show the benefits of the procedure. Data sets were executed on every processor in the system, simultaneously and reduction in time lapse was shown in this chapter. Reduction in time lapse makes the parallel processing useful for data execution.

Relatively cheap and less complex processors was used for analysis because large data set can be divided to every processors under the parallel processing procedure and consequently, workload on the each processors are drastically decreased. Data sets, which were stored on the related table, were used to show effect of the processors parallel performance. Their values were controlled under limitations individually. They were divided to available processors and process of evaluation of the data was completed in a short period of time. Results were presented to emphasize the reduction of elapsed time using parallel processor.

4.1 Measurement Results of Remote Monitoring System

In 18.01.2017, input and output voltage and current of MPPT, voltage of each battery, temperature, irradiation, dust density, humidity are measured. Variation of these quantities can be investigated using the data shared in the graphs. Measurements were taken in 30 second time interval. Sensor, which was used to measure temperature, was located near batteries and MPPT. The reason of temperature increment was considered because of air conditioner. Air conditioner was kept to heat the room. Temperature was also measured by another sensor and results were verified. Temperature measurements are shown in Figure 4.1.

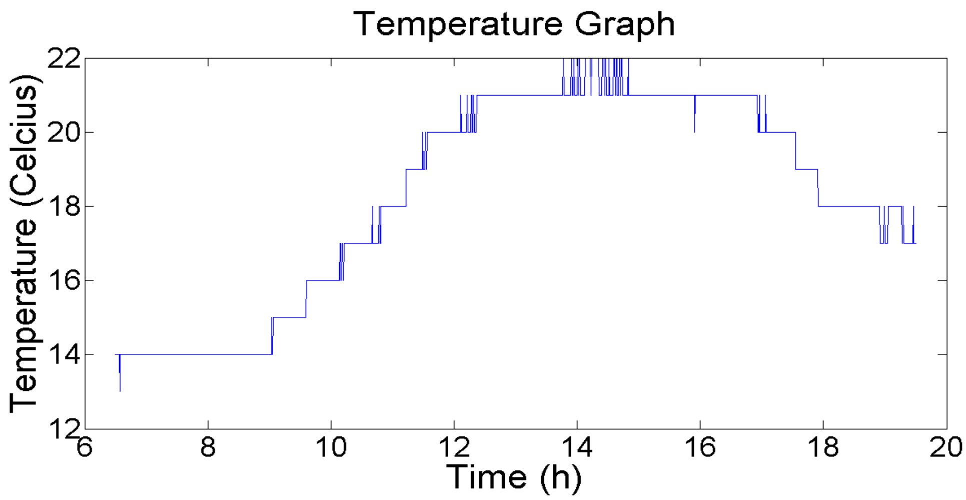


Figure 4.1 Temperature(C) versus time graph(Measurements are taken in 18.01.2017)

MPPT, batteries and humidity sensor were located in same ambient. Variations of humidity were observed between 35 and 39. Instant decrease and increase can be seen in both temperature and humidity readings. Resolution of these sensors was one. Humidity readings are shown in Figure 4.2.

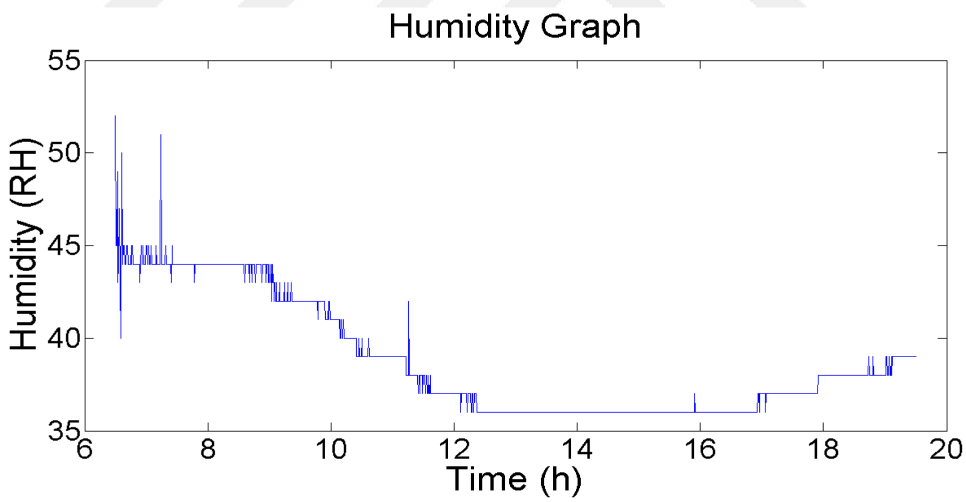


Figure 4.2 Humidity(rh) versus time graph(Measurements are taken in 18.01.2017)

System does not involve a load except batteries. Batteries are fully charged long before the measurements. A spindle was connected to the photovoltaic system as a load. Spindle is a part of computer numerical control(CNC) machine and it is used for carve object from bulk material. CNC supply was different than photovoltaic system. Only the spindle was connected to the photovoltaic system to observe current measurements. Thus, current readings are changing based on spindle working conditions. CNC was carve wood during operations. Fluctuations of input and output current of the MPPT was depend on current situation of spindle. Input and output currents readings are shown in Figure 4.3 and Figure 4.4.

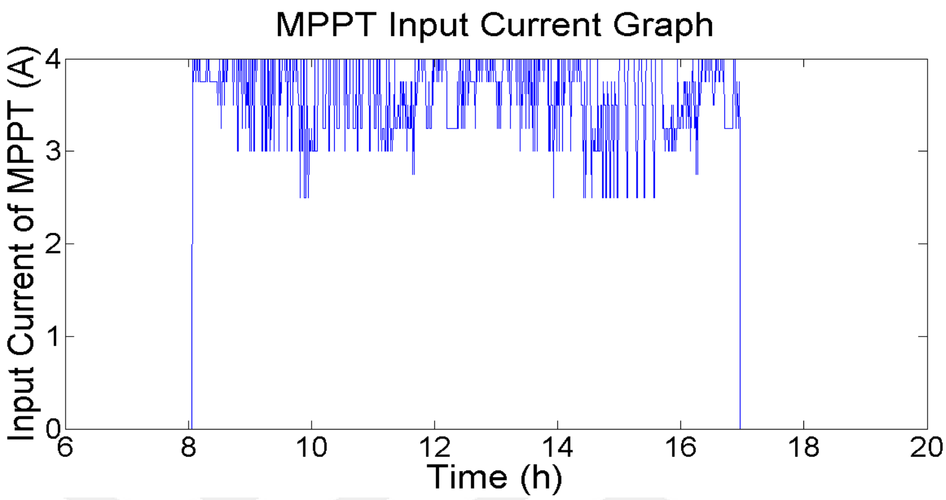


Figure 4.3 Input current of MPPT(A) versus time graph(Measurements were taken in 18.01.2017)

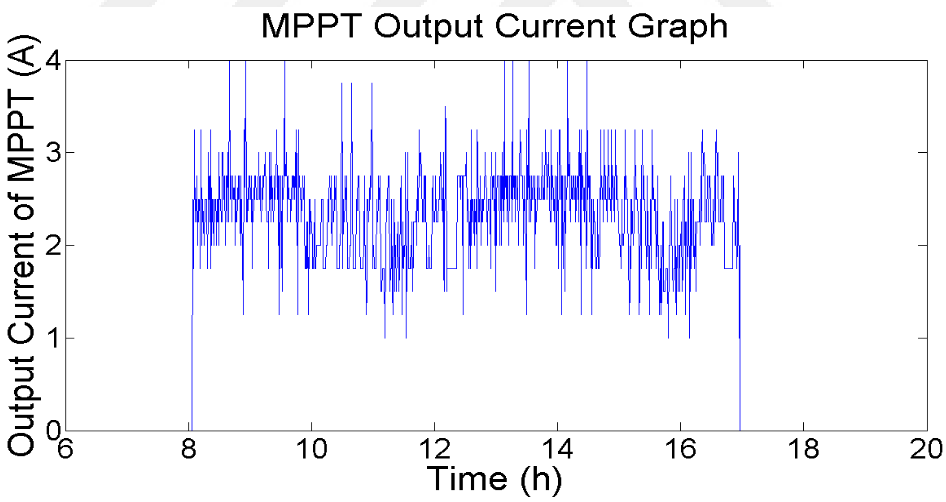


Figure 4.4 Output current of MPPT(A) versus time graph(Measurements were taken in 18.01.2017)

Input and output voltage of the MPPT taken using voltage dividers. Output voltage of the MPPT was connected to batteries. Output voltage of the MPPT was kept constant. However, input voltage of the MPPT was changing +-4V during the mid of the day. At night, input voltage of the MPPT was decreased up to 30 V. When input voltage of the MPPT was lower than output voltage of the MPPT, current flow was stopped from photovoltaic panels output to spindle. Spindle was closed to prevent charging sequence of batteries. Batteries are never used and they have been fully charged for almost two years. Input and output voltage readings are shown in Figure 4.5 and Figure 4.6.

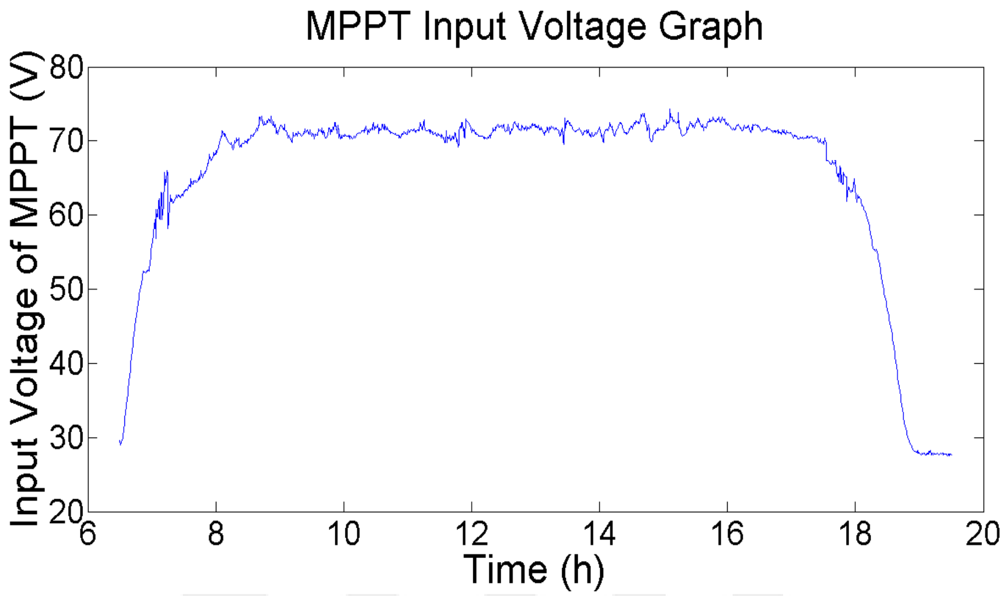


Figure 4.5 Input voltage of MPPT(V) versus time graph(Measurements were taken in 18.01.2017)

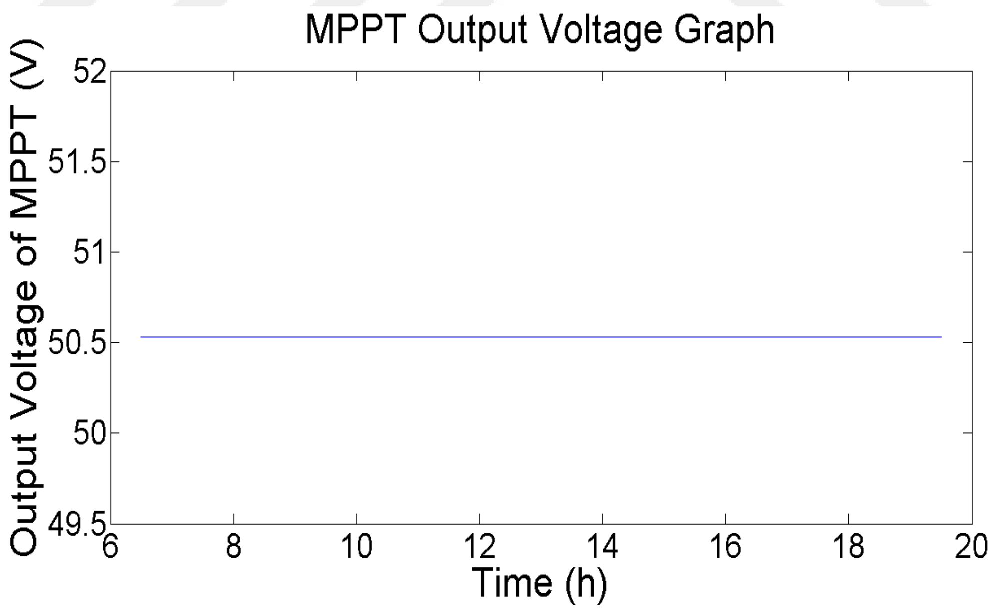


Figure 4.6 Output voltage of MPPT(V) versus time graph(Measurements were taken in 18.01.2017)

Voltage of each battery was same because they are fully charged and connected series to each other. Load does not exist in the system for batteries. Thus, stored energy has not been consumed for almost two year. Furthermore, summation voltage of each batteries was obtained same with output voltage of the MPPT. Voltage of each battery was shown in Figure 4.7, Figure 4.8, 4.9 and Figure 4.10

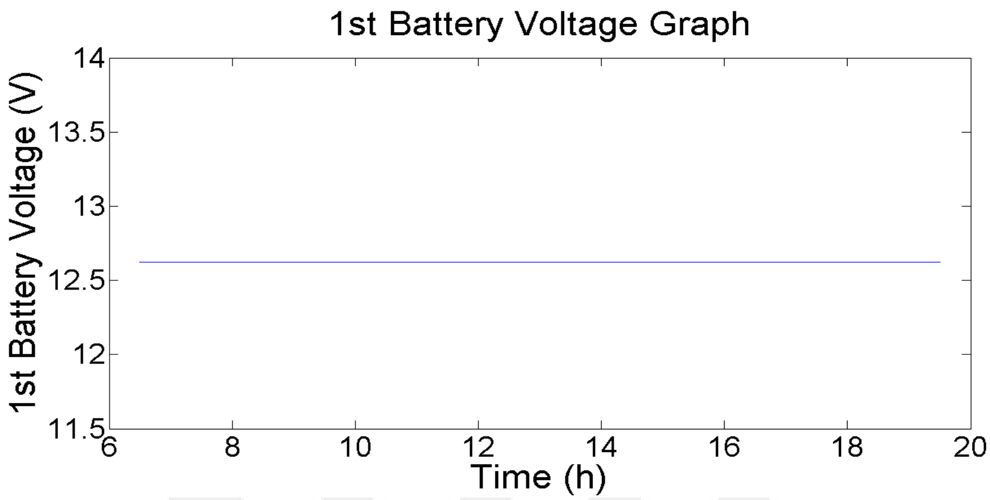


Figure 4.7 First battery voltage(V) versus time graph(Measurements were taken in 18.01.2017)

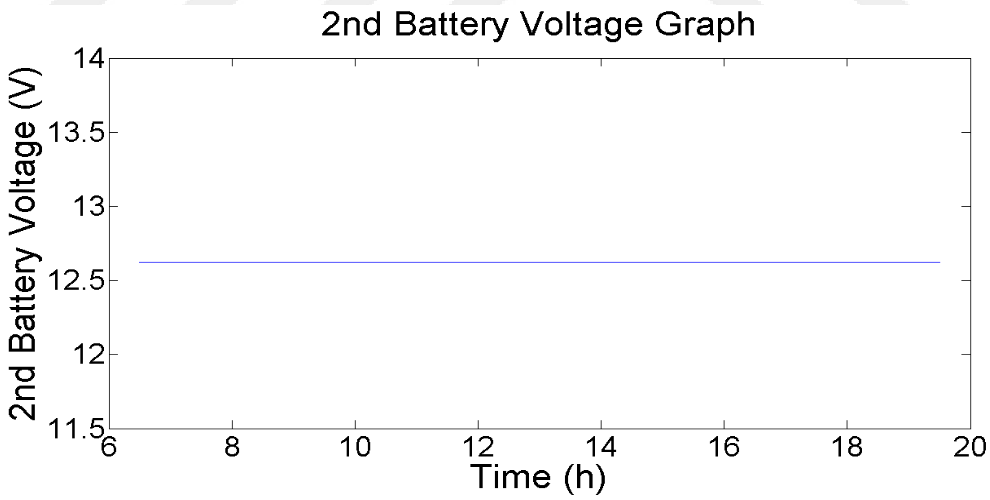


Figure 4.8 Second battery voltage(V) versus time graph(Measurements were taken in 18.01.2017)

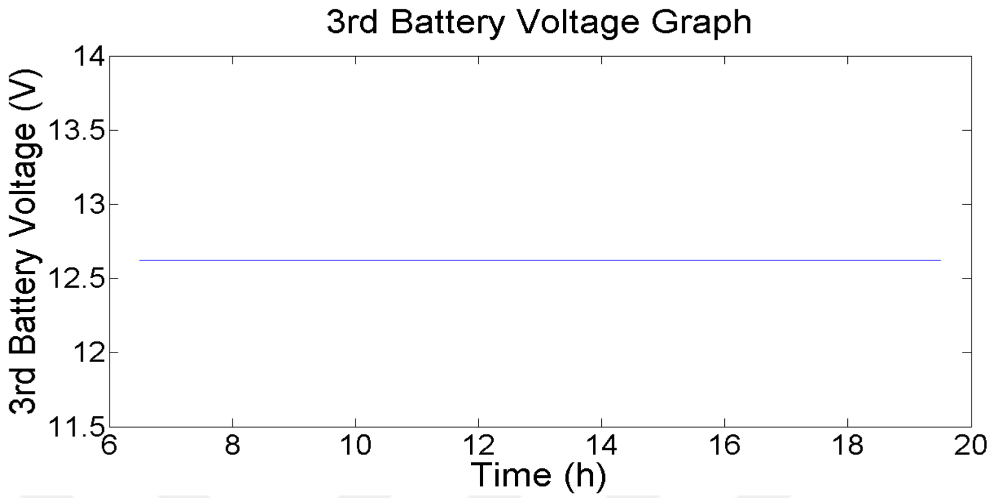


Figure 4.9 Third battery voltage(V) versus time graph(Measurements were taken in 18.01.2017)

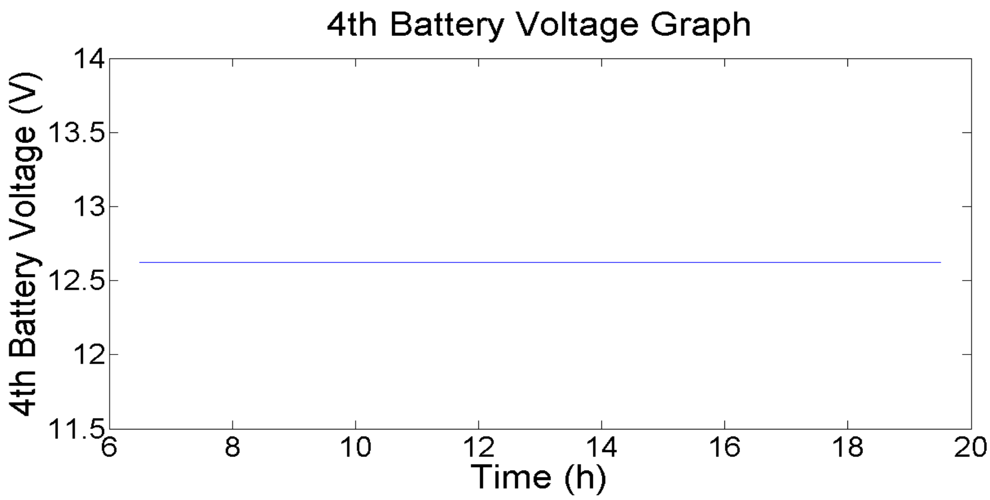


Figure 4.10 Fourth battery voltage(V) versus time graph(Measurements were taken in 18.01.2017)

Dust sensor is located in the same ambient with MPPT and batteries. Output of the sensor is fluctuated between 50-280 mg/m3. The difference among the measured data were considered due to the effect of CNC. Curve operations of CNC is caused to increase of dust density. Dust densities are shown in Figure 4.11.

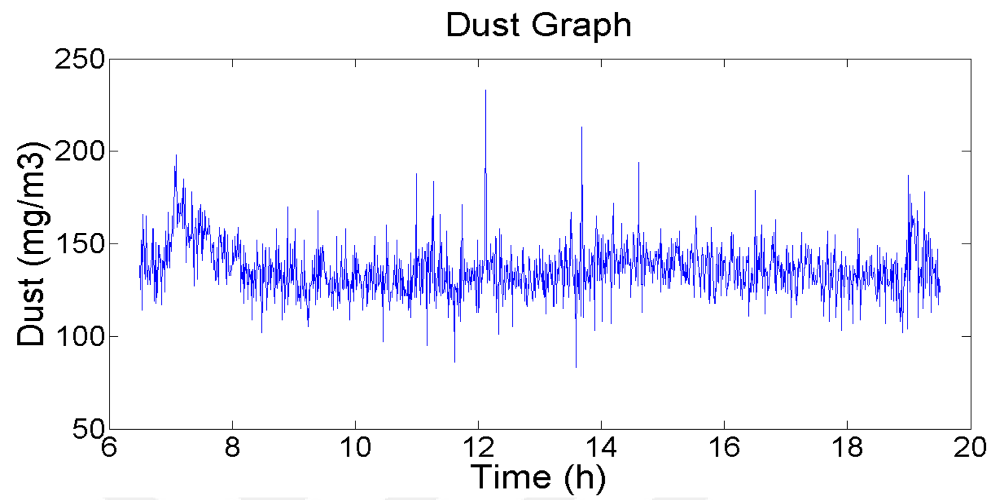


Figure 4.11 Dust density (mg/m3) versus time graph(Measurements were taken in 18.01.2017)

Finally, Irradiation is measured using reference cell and it is located closed to the photovoltaic panels with same angle. Peak readings were taken in the mid of the day and as expected, irradiation was drastically decreased at the night. Irradiations of Dokuz Eylül University in 26.02.2017 from mid of the day to night(06.45-19:45) are shown in Figure 4.12.

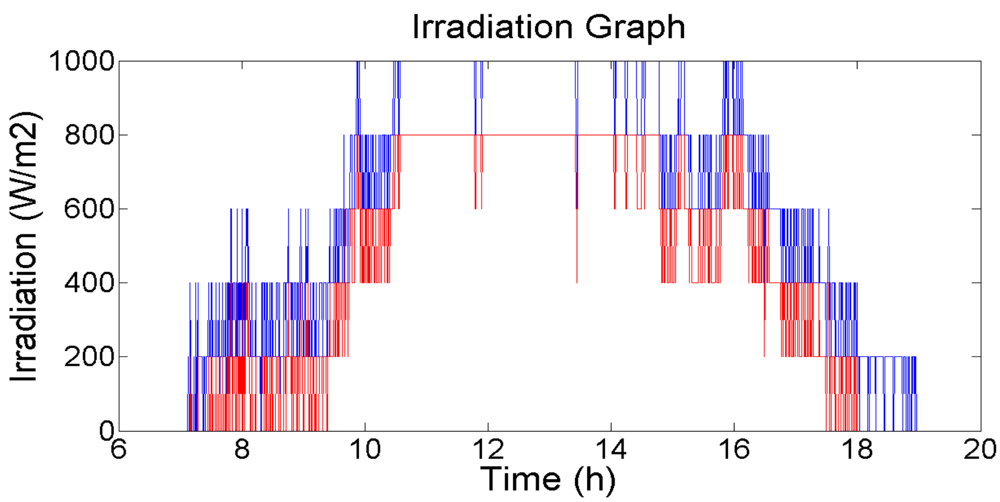


Figure 4.12 Irradiation(W/m2) versus time graph(Measurements were taken in 18.01.2017)

Current sensor output is changing between 2.5-5 V. Each 1 A change in the input is increased the output 60 mV. However, PIC is able to measure the voltage with 50 mV resolution. Hence, the resolution of related irradiation measurement is 200 w/m2 as can been 4.13. Instant peaks or drops caused by the shadows is also eliminated in Figure 4.13.

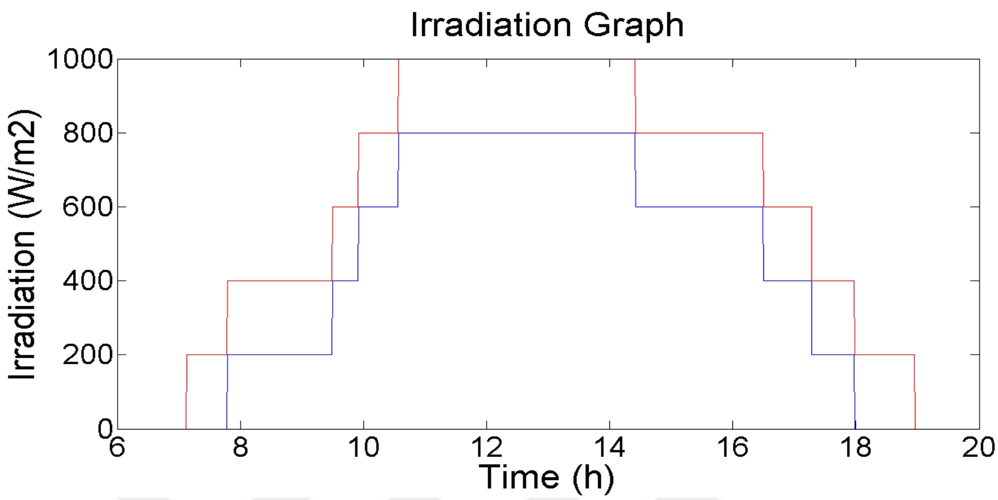


Figure 4.13 Irradiation graph with upper and lower boundaries(Measurements were taken in 26.02.2017)

This resolution of 200 W/m2 restricts the resulting curve to have smooth transition. The upper and lower boundaries of the measurements 0-100 W/m2. Hence, the results can only have 5 different steps with this resolution. In order to get a better irradiation curve, a smoothing process is applied to the system by plotting a curve between the upper and lower boundaries of the measurement. This approach allow us to see how the result would be if a better hardware that can give better resolutions was used(Figure 4.14).

4.2 Data Visualization

The system is designed to remotely observe the measured data for the sake of simplicity. Thus, website and graphical user interface was designed. Instantaneous measurements can be observed and downloaded using website. The graphs were plotted by taking the last 15 observed measurement for each data sets. Furthermore, graph of last 15 measurements can be also seen in GUI. Additionally, past measured

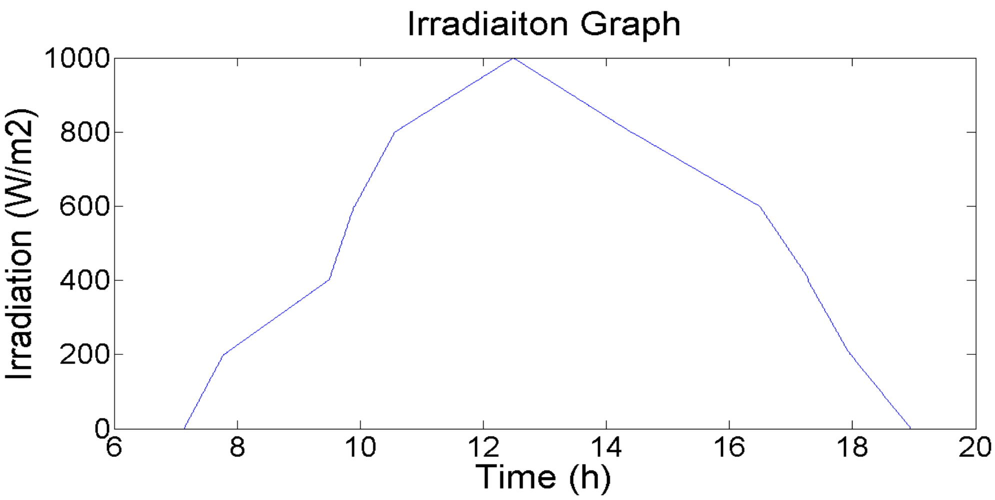


Figure 4.14 Smoothing applied irradiation curve (Measurements were taken in 26.02.2017)

data can be controlled with parallel processing principle using the GUI.

4.2.1 Graphical User Interface

A design was realized with GUI to observe results. Measurement was initiated using related buttons. It was designed for only two remote stations but it can be updated using same methods and codes for more than two remote station if they are necessary for the system. One of the remote stations was completed with measurements but other station is still available for parallel processing. Main page of the GUI includes graphs, test and start measurements features which are shown in Figure 4.15. Temperature measurements of remote station is depicted in in Figure 4.15.

Temperature of first remote station was obtained 15 Celcius degree by two different sensors in each measurement. Temperature of center station was measured by second remote station to ensure communication. Temperature of second remote station was found 15 Celcius degree by two different sensors in each measurement. Temperature measurements were performed at least 8 times to verify the measurement reliability. The standard deviation was found less than 1 % degree which is negligible for measurement systems. Temperature and humidity data were used to test the accuracy (or quality) of MPI communication between remote and the center stations. If the MPI communication test was successfully completed, the

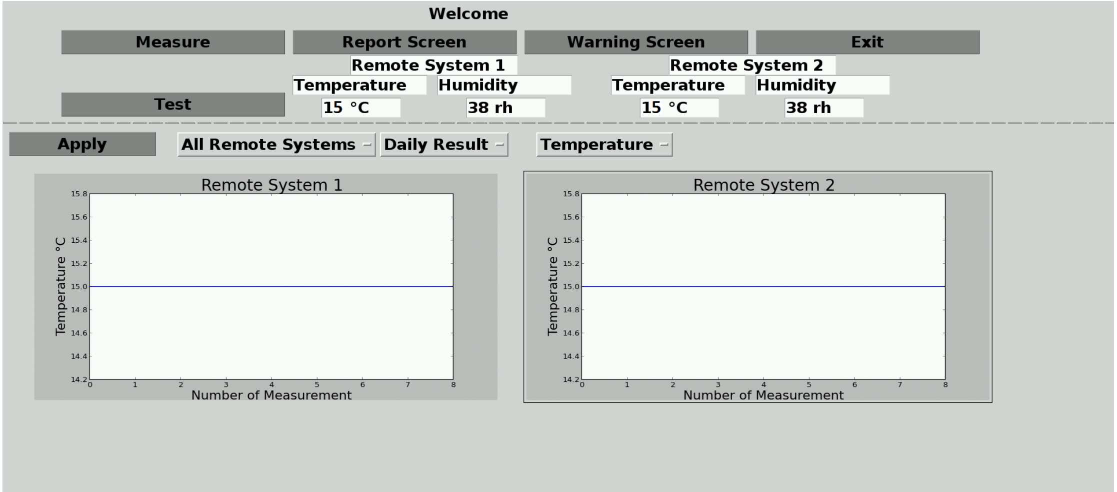


Figure 4.15 Main page of GUI with graphs on the seven inch TFT LCD display

textviews, which were located to visualize incoming data under the Remote System 1 and Remote System 2, are filled with related numbers(Figure 4.16). Temperature is measured 15 Celcius during test.

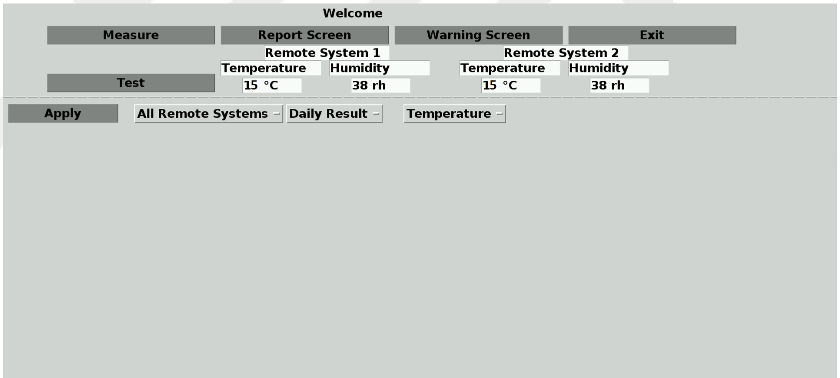


Figure 4.16 Results of the remote and the center stations' communication test

There is a difference between main page result and report screen result in terms of result visualization. Main page results were shown in graphs but report screen results were given in the boxes with numbers. All results are visualized together under the third option menu in order to make the results more comprehensive and understandable at first glance. All results were shared via third option, which is shown in Figure 4.17.

The order of parallel computing was organized using warning screen. Measured data, which will be investigated, were selected using both edit text or option menu. After data selections were identified, all the columns, which is related the selection

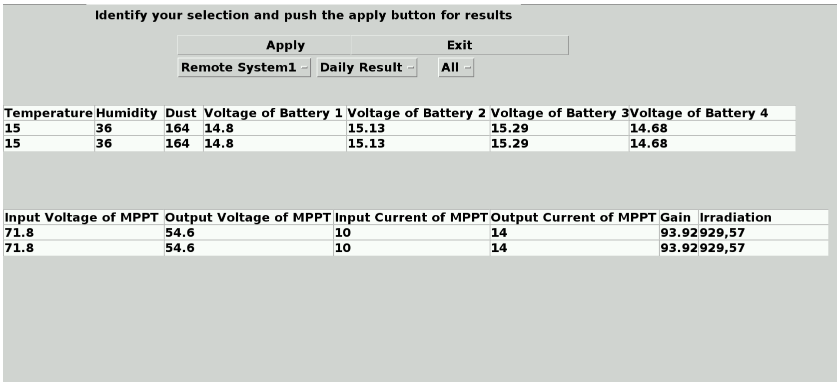


Figure 4.17 All measurement chosen results of the report screen

part, were taken from database.

Data were distributed over MPI based on number of columns to achieve equal working load condition. These data were observed under predefined limits. Then mismatched ones were sent back to the center station. Center Raspberry Pi visualizes the data as shown in Figure 4.18. I mostly did not observe mismatch data during the measurements.

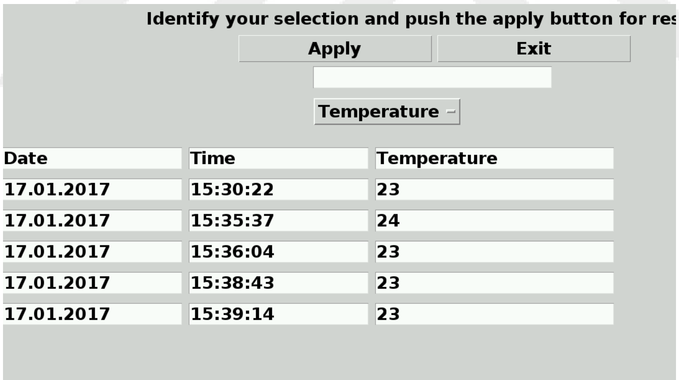


Figure 4.18 Mismatched results after parallel processing on warning screen

4.2.2 Website

As explained in chapter three, website has been designed to remotely observe measurements. Login protected data can be observed using website. Data page of the website was designed to take feedback current measurements and visualize daily graphs. JPgraph, which is established as a library for graphs, was used for visualization. JPgraph library option should be installed at the beginning of the process. Installed data must be stored in the same location and they should be imported as a library by calling it as functions. After downloading of measurement, data taken from related column can be saved as an image with the desired format and the images can be added on the website using HTML tags.

Page refreshing was created when login call or page refreshing comes and new images of graphs were saved with the same method and names. Therefore, previous images are not stored on Raspberry Pi because storage is limited with eight (8) GB SD card. The last measured data for each one was illustrated using simple text viewer. All measurement graphs can be seen using related button. They can be seen in Figure 4.19. Temperature and humidity results are given in Figure 4.19. Measurements were taken 5 minutes interval for the last 15 measurements, which corresponds to hour 15 minutes, are shown in Figure 4.19. Thus, readings are not changed during 1 hour 15 minutes. Temperature and humidity did not change during whole measurements. Standard deviation among the number of measurements were calculated less than 1 %. It justifies the robustness of developed technique in this study.

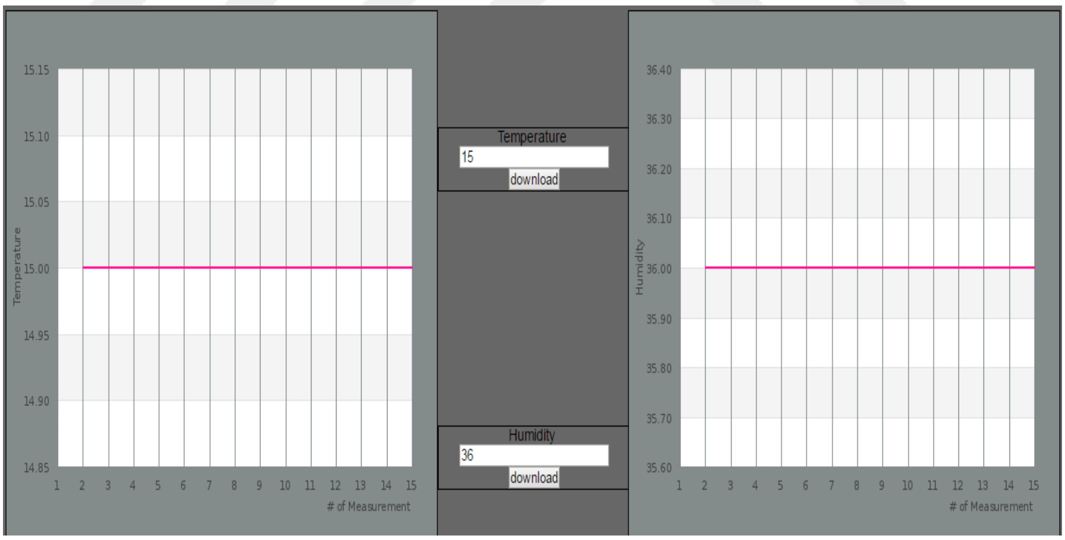


Figure 4.19 Graphs and last measurements of voltage battery 3 and 4 in website

Every data can be downloaded using button located under all graphs individually in a day period. Downloadable content was defined in xls (excel) format but time interval and format can be changed like txt-doc regarding to necessity of user. GUI should be used for detailed download.

4.3 Parallel Performance

To investigate the parallel performance, four tests were realized and results were compared to each other. These four tests are designed using same procedure with same processes and only number of processor was changed. Process was defined as control of the pre-measured value using predefined limitations. These tests involve temperature, humidity, dust and voltage information taken from solar plant. MPI embedded Python code was used to apply the parallel processing procedure. First part of the program executes the Python-MYSQL database interaction. Data type (voltage, temperature) is the first necessary input and this information is directly taken from GUI. User name and passwords of database are other inputs for tests. Column of the table and database name was identified using these inputs and data inside of the columns can be investigated using user name and password.

Transmission of the data with MPI was done running the second part of the code. Working load was expressed as a division of data length to processors count. Division of data length is not the solution of equal working load problem to investigate the real performance of parallelism. Remain part of the division was assigned and added the center station Raspberry Pi. This procedure generally causes to scale up fraction of the code whereas it can be ignored in our study because number of processes are high enough to neglect the effect of scale up fraction.

The first test was conducted with serial processing in order to compare the performance of parallel processing. Data pool of the test, which are taken from related table of database, array of 10000 different measurements. Every data was investigated using a processor. Second, third and fourth tests are designed to complete the same process after data pool were divided to remote stations. The number of processors was increased in every test. Clusters of Raspberry Pi consisting of one, two, three, four single board computers were used to realize the test by processors. Division of the 1000 data to three processors extract the biggest remain as 3. Three data set send to processors and 3 as a remain add to center processor workload. Therefore, remain of division is always under 3 and 10000 readings are investigated in every test. Comparison of the data pool and remain part was revealed the error less than 1 %. The influence of the work load on parallelism was also found less than 0.1%. However, fraction of the code was increased by work load problems. On the other hand, fraction of the code was calculated using Karp-Flatt metric to investigate the test results.

Transmitted data were checked by monitoring the results for possible incorrect results. Limits were specified with respect to the specifications such as maximum battery temperature. Inconsistent data was sent back to center station Raspberry Pi and the third step of the code was completed. Result data were visualized on GUI in the last part of the code. However, speeding up data must be presented to convince users to show necessity of parallel processing.

Results for investigation of measurements can be seen in Table (4.1). These results involve the outputs of the tests for Clusters of Raspberry Pis consisting of one, two, three, four single board computers as processors. Elapsed time throughout the tests was identified as an output to understand the speed up. The number of processor used in the test can be noticed using test number. It is obvious that the increment on the number of processor is decreased the required time to complete the measurements.

The averages of whole results were calculated to indicate the enhancement of the performance of parallel processing increasing the number of processors. Fifty (50) tests were applied at different time during the day in order to detect the background effects and time delays due to the possible internet connection errors. Generally, background tasks increase the time lapse. However, elapsed time can be decreased or increased according to internet connection speed. The average time (s) needed for each measurement was given in the Table (4.1). Average of elapsed time and standard deviation are calculated 0.483118858-0.03333687 for 1 processor, 0.246761448-0.01306581 for 2 processors, 0.167411032-0.02028203 for three processors, 0.128194851-0.02753782 for four processors, respectively. The highest error was obtained at most 8% for 3 Processor. The results are very convincing to show the reliability of test system.

Table 4.1 Elapsed time(second) list of the pre measured information with different number of processors(50 same test were applied to observe effects of connection problems or background tasks)(Second is abbreviated as S)

Test Number	1 Processors(S)	2 Processors(S)	3 Processors(S)	4 Processors(S)
1	0.48165106	0.24578196	0.17409706	0.12932395
2	0.48110389	0.25356911	0.17490696	0.13007903
3	0.49645900	0.24867698	0.16868305	0.12882100
4	0.49468507	0.24497699	0.17211198	0.12845604
5	0.48733688	0.24707198	0.16118598	0.15175390
6	0.47444200	0.25228405	0.16706109	0.12853813
7	0.48153495	0.24593494	0.16852593	0.14832091
8	0.47385001	0.25692006	0.17483496	0.12839385
9	0.48457789	0.23924303	0.17485404	0.12868905
10	0.47119689	0.24773502	0.16232395	0.12688607
11	0.47745990	0.24881291	0.17792105	0.12740016
12	0.48368191	0.24879097	0.16439700	0.12815687
13	0.48925709	0.24694490	0.17185401	0.12627985
14	0.48148012	0.24899196	0.16203212	0.12868905
15	0.48665905	0.24448704	0.16738200	0.12804312
16	0.47358608	0.24819803	0.16191005	0.12813687
17	0.47967219	0.24947500	0.16466093	0.12825608
18	0.47903609	0.24623394	0.16194891	0.12864286
19	0.47214388	0.24636197	0.16454601	0.12632513
20	0.48460793	0.24723696	0.16649198	0.12690687
21	0.50287985	0.24536204	0.16622614	0.12520599
22	0.46954298	0.24524593	0.16675305	0.12674903
23	0.48832795	0.24645400	0.16954994	0.12638931
24	0.48115396	0.25231099	0.16369891	0.12654900
25	0.47637200	0.24511501	0.16848397	0.12596488

Table 4.1 Elapsed time(second) list of the pre measured information with different number of processors(50 same test were applied to observe effects of connection problems or background tasks)(Second is abbreviated as S) (cont.)

Test Number	1 Processors(S)	2 Processors(S)	3 Processors(S)	4 Processors(S)
26	0.48312401	0.24537894	0.17163109	0.12839388
27	0.48381805	0.24510312	0.16330695	0.12872602
28	0.47305703	0.24093890	0.16155982	0.12800197
29	0.48061513	0.24902080	0.16496491	0.12841320
30	0.48030901	0.24014306	0.16450214	0.12811295
31	0.47718882	0.24179101	0.16103196	0.12507391
32	0.47992706	0.24547791	0.16519188	0.12630605
33	0.48684716	0.24573087	0.16496491	0.12854418
34	0.48463511	0.25853395	0.17238402	0.12542605
35	0.48202288	0.25194286	0.7158913	0.12504816
36	0.48327803	0.25082611	0.16459798	0.12529492
37	0.48783302	0.24271798	0.16810393	0.12652492
38	0.48805186	0.24447488	0.17046499	0.12815308
39	0.48961301	0.24300217	0.16489887	0.12834901
40	0.48659491	0.24806419	0.16382384	0.12421608
41	0.49939417	0.23887705	0.16492080	0.12697696
42	0.48019599	0.24228286	0.16793203	0.12632012
43	0.49534894	0.25660181	0.16092014	0.12807092
44	0.47713494	0.24098587	0.16267108	0.12627482
45	0.48573994	0.24098205	0.16658115	0.12795305
46	0.48742222	0.24209702	0.17361187	0.12490391
47	0.48686790	0.25109996	0.16479587	0.12588810
48	0.48215007	0.24719203	0.16587090	0.12834600
49	0.47396993	0.25160899	0.17247605	0.12756705
50	0.48810482	0.24098205	0.18131399	0.12590003

According to Amdahl law, the speedup was expected to increase by increasing number of processors. Both theoretical and experimental results were shared in the Table (4.2). The highest difference between actual and Amdahl's Speedup was calculated 0.14 %. The results were obtained consistent with theory. Same tests were conducted many times to calculate the fraction of the code. Fraction of the code is an another parameter to show the performance of parallel processing. Fraction of the code was calculated using Karp-Flatt metric as 2%. The fraction of the code results obtained by Karp-Flatt technique was also confirmed t he r esults o btained with Amdahl rules and it is consistent with test results.

Table 4.2 Effect of processor addition on speed up. Fraction of the code is calculated as 2%

Number of Cores	Mean of Results(Seconds)	Actual Speedup	Amdahl's Speedup
1	0.483118858	1	1
2	0.246761448	1.9578	1.9607
3	0.167411032	2.8858	2.8846
4	0.128194851	3.7686	3.7735

Communication is the major plan to use MPI and the parallel processing was constituted to choose the program. As a result of the method, processor which was preferred for organizing the whole system, was relatively chosen cheap and less developed. Cost of the system is the one of the most important parameter for every design. University campus and factories are targeted regions for the design because both of them involve same internet connection. The connection of the processors must be same to realize communication for parallel processing.

CHAPTER FIVE
CONCLUSION

Remote monitoring of photovoltaic system for 1kW powered plant has been designed and realized. A center station and a remote station are completed. Input and output voltage and current of MPPT, voltage of each battery, temperature, irradiation, dust density, humidity and efficiency of MPPT have been measured or calculated. Single board computers are used to organize the measurements and communication between them is accomplished using MPI.

Single board computers must be connected in the same internet connection to use MPI. Parallel processing of the remote system's related operations were also executed using MPI. Remotely monitored measurements can be visually observed using GUI in center station and thesis website.

Measurements are critically important for photovoltaic plants and investigation of their connected devices' performance. These readings and readings related research have been completed relatively low cost with high efficiency. Design has been done to use it in area which exists same connection such as university campuses and factories.

Designed system was implemented in Dokuz Eylül University Tınaztepe Campus. It can be integrated in low powered systems using same algorithm and procedure. Parallel processing is the key feature of the study and performance increase was also presented using tests, which will always use to control measured data. These tests were also added the GUI in warning screen to control past measurements. Therefore, relatively low cost processors can be used to organize and visualize the measurements.

One of the other benefits of thesis is remote station addition to the system. SD card on the remote system can be copied for extra remote station. It will ready for measurement and communication. Center station code should be modified using same procedure and table of extra station must be created to use it.

REFERENCES

Al-Dahoud, A., Fezari, M., & Belhouchet, Z. B. (2014). Remote monitoring system using wsn for solar power panels. *IEE-SIMS '14 Proceedings of the 2014 First International Conference on Systems Informatics, Modelling and Simulation*, *15*, 149–154.

Amdahl, G. M. (1967). Validity of the single processor approach to achieving large scale computing capabilities. *IEEE Solid-State Circuits Society Newsletter*, *12*, 19–20.

Annavaram, M., Grochowski, E., & Shen, J. (2005). Mitigating Amdahl's law through epi throttling. *Proceedings of the 32nd International Symposium on Computer Architecture IEEE*, *36*, 1042–1052.

Backus, J. (1978). Can programming be liberated from the von neumann style?: a functional style and its algebra of programs. *Communications of the ACM*, *21*, 613–641.

Bagnasco, A., Allasia, G., Giannettoni, M., Pinceti, P., & Parodi, G. (2012). Innovative solutions for photovoltaic plants remote monitoring. *IEEE*, 1 – 5.

Chorley, M. J., & Walker, D. W. (2010). Performance analysis of a hybrid mpi/openmp application on multi-core clusters. *Journal of Computational Science*, *1*, 168–174.

Cox, S., Cox, J., Boardman, R., Johnston, S., Scott, M., & O'Brien, N. (2014). Iridis-pi: a low-cost, compact demonstration cluster. *Cluster Computing*, *17*, 349–358.

Energy information administration statistic (2016). Retrieved 30 July 2016, from http://www.eia.gov/cfapps/ipdbproject/iedindex3.cfm.

Fang, J., & Wang, F. (2011). Design of green house remote monitoring system based on labview. *IEEE-Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference*, *1*, 536–539.

Flynn, M. J. (1972). Some computer organizations and their effectiveness. *IEEE Transactions on Computers*, *C-21*, 948 – 960.

Gabriel, E., Fagg, G., Bosilca, G., Angskun, T., Dongara, J., & Squyres, M. (2004). Open mpi: goals, concept, and cesign of a next generation mpi implementation. *Proceedings, 11th European PVM/MPI Users' Group Meeting*, 97–104.

Hill, M., & Marty, M. (2008). Amdahl's law in the multicore era. *IEEE Computer*, *41*, 33–38.

Karp, A. H., & Flatt, H. P. (1990). Measuring parallel processor performance. *Communications of the ACM*, *33*, 539–543.

Köklü, T., & Kılınç, S. (2016). Remote monitoring of photovoltaic systems using embedded system clusters. *24th Signal Processing and Communication Application Conference(SIU)*, Zonguldak.

Köklü, T., Tamer, O., Öztürk, R., & Türkoral, T. (2015). Parallel performance research of single board computers. *The Chamber of Electrical Engineers-İlk Bildiriler Konferansı*, Ankara.

McKinney, G., & West, J. (1993). Multiprocessing mcnpon an ibm rs/6000 cluster. *1993 Annual Meeting of the American Nuclear Society,*.

Nunes, R., & Almeida, J. (2010). Parallelization of sequential gaussian, indicator and direct simulation algorithms. *Computer Geoscience*, *36*, 1042–1052.

Papadakis, K., Koutroulis, E., & Kalaitzakis, K. (2005). A server database system for remote monitoring and operational evaluation of renewable energy sources plants. *Science Direct*, *30*, 1649–1669.

Rashidi, Y., Moallem, M., & Vojdani, S. (2011). Wireless zigbee system for performance monitoring of photovoltaic panels. *Photovoltaic Specialists Conference 2011 37th IEEE*, 3205 – 3207.

Raspberrypi GPIO (2016). Retrieved 31 July 2016, from https://www.raspberrypi.org/documentation/usage/gpio/.

Renewable report (2016). Retrieved 10 June 2016, from https://yearbook.enerdata.net/wind-solar-share-electricity-production.html#renewable-in-electricity-production-share-by-region.html.

Solar energy in Turkey (2016). Retrieved 30 July 2016, from http://www.eie.gov.tr/MyCalculator/Default.aspx.

Tejwani, R., Kumar, G., & Solanki, C. (2013). Remote monitoring for solar photovoltaic systems in rural application using gsm voice channel. *Energy Procedia*, *57*, 1526—-1535.

Turkey renewable law (2016). Retrieved 31 July 2016, from http://www.eud.org.tr/EN/Genel/Default.aspx?17A16AE30572D313AAF6AA849816B2EF4376734BED947CDE.

Velde, G., Bickelhaupt, F., Baerend, E., Guerra, C., & Van Gisbergen, S. (2000). Chemistry with adf. *Journal of Computational Chemistry*, *22*, 931–967.

Xiaoli, X., & Daoe, Q. (2011). Remote monitoring and control of photovoltaic system using wireless sensor network. *IEEE-Electric Information and Control Engineering International Conference*, 633 – 638.

Yavits, L., Morad, A., & Ginosar, R. (2013). The effect of communication and synchronization on amdahl's law in multicore systems. *Parallel Computing*, *40*, 1–16.