Implementation of IoT based Dual Axis Photo-Voltaic Solar Tracker with MPPT using Raspberry-Pi and NodeMCU

Mr. A M Veera Sai[1], Dr. K Rama Sudha[2]

[1]M. Tech Student, Department of Electrical Engineering, Andhra University College of Engineering, Andhra Pradesh, India.

[2]Professor, Department of Electrical Engineering, Andhra University College of Engineering, Andhra Pradesh, India

---***---

Abstract – *In this study IoT (Internet of Things) based Dual Axis Photo-Voltaic (PV) solar tracker with maximum power point tracking (MPPT) is proposed. In order to extract more amount of energy from PV module, MPPT technique along with calculation of azimuth angle and sun elevation angle are done. So that we can obtain the position of the sun with respect to the PV panel geographical location. Unlike sensor-based PV tracker which uses auxiliary devices such as LDR, small PV cells, photo diodes, etc, to obtain the position of the sun, here we have used sun equations to obtain the sun's position taking input parameters as latitude, longitude, time and day number. MQTT (Message Queuing Telemetry Transport) protocol is used to transmit and receive data form PV module (Node-MCU) and server (Node-RED). The shared information will be voltage, current, azimuth angle, altitude angle and duty cycle for boost converter. Here Algorithm for solving sun equations and MPTT logic is placed in Node-RED, it calculates the duty cycle required to extract the maximum power from the PV system. Node-RED platform is installed on the Raspberry-Pi in which mosquito broker is installed, and for changing the position of PV module 2 servo motors were used and one Node-MCU which acts as a client for the server and also it acts as a controller for the PV module and boost converter. A photovoltaic system has been constructed and experimentally verified that the proposed sun tracker finds the sun's position with a tracking error of 0.1^o and 0.47^o for altitude and azimuth angle respectively, and there is an increase in energy of 31.92%, using proposed sun tracker.*

Key words: IoT, Node-RED, NodeMCU, MPPT, Photovoltaic system.

1. INTRODUCTION

IoT (Internet of Things) is defined as "A network of internet connected objects able to collect and exchange data". The IoT is influencing our lifestyle from the way we react to the way we behave form air conditioners that you can control with your smartphone to smart cars providing the shortest route or your smart watch is tracking your daily activities IoT is a giant network with connected devices[1]. These devices gather information and share data about how they are used in the environment and how they are operated it's all done using sensors, sensors are embedded in every physical device which can be traffic lights, smart watches, etc that we come across in day to day life [3]. These sensors continuously emit data about the working state of the devices, but the important question is how they share this huge amount of data, and how do we put this data to our benefit. Internet of Things provides a common platform for all these devices to dispose their data and a common language for all these devices to communicate with each other. Various sensors which emit data are sent to IoT platform security. IoT platform integrates the collected data from various sources further analytics is performed on the data and valuable information is extracted as per requirement [4]. Finally, the result is shared with other devices for better user experience automation and improving efficiencies. The future of IoT industry is huge. Business insider intelligence estimates that around 24 billion internet connected devices will be installed by 2020[2].

Node-RED is a powerful tool for building IoT applications with a focus on wiring code blocks or sensor blocks, known as nodes, together to perform a task[5]. Node-RED is a flow-based programming development tool for visual programming. Originally it was developed as an open source project at IBM to meet their need to connect their hardware and devices to web service and other software, but now it has become as an essential tool for IoT programming. Node-RED was pre-installed in Raspberry-Pi, SIMATIC IoT 2020, Artik, etc. Using Raspberry-Pi, we can use Node-RED[6] and access the node-red ui from the browser using (raspberry-pi ip address):1880[7].

MQTT (Message Queuing Telemetry Transport) is a messaging protocol that works on the principle of publish and subscribe mechanism, for simply communicating between the machines. For running it is required to install a server which is named as broker. We can install the broker on almost all platforms (windows, mac os, linux…) and on raspberry-pi. Here mosquito broker [8] is installed on Raspberry Pi. MQTT consumes 11 times less energy compared to HTTP for transferring messages [9].

Node-RED dashboard is a best tool for controlling monitoring our IoT application. It is a user interface for all our flow which was designed on node-red docker.

2. ANALYSIS OF PROPOSED TRACKER

2.1 Earth – Sun equations:

First, we must know the location of the sun relative to the collection device in order to collect energy from sun. Unique vector approach is used to obtain the sun equations[10]-[12]. Let us start finding Earth-Sun Angles (hour angle, declination angle, latitude angle). As we know earth revolves around the sun in an elliptical orbit in 365.25 days with a distance of $1.496*10^{11}$m(1 AU) from sun and on aphelion(3rd of July) earth reaches its maximum distance form sun, and minimum distance on perihelion(2nd of Jan) over the ecliptic plane.

Earth is inclined with polar axis by 23.45^0. The tilt of this axis produces seasons with revolving around the sun. Time is measured with rotation of the earth about its axis, taking Greenwich, England and poles which is known as Prime Meridian. Since earth's orbit around the sun is elliptical, it travels faster when it is nearer to sun, due to which we have two times solar time and standard time. In solar time solar noon occurs at the instant when sun positioned is due south with relative to the geographic location, there is a difference of around 44minutes between solar and standard time.

Hour angle (ω) is described as the distance between the meridian of the observer and meridian in which sun rays are parallel. $\omega = 0$ on solar noon and it is increased by 15° for every hour.

$$\omega = 15(t_s - 12) \text{ degrees} \tag{1}$$

where t_s = solar time in hours.

Equation of time (EOT) is defined as the difference between mean solar time and true solar time on a give date. EOT is determined from the below equation which was developed by Woolf (1968) which have a accuracy of around thirty seconds.

$$\text{EOT} = 0.258\cos x - 7.416\sin x - 3.648\cos 2x - 9.228\sin 2x \text{ minutes} \tag{2}$$

$$X = \frac{360(N-1)}{365.242} \text{ degrees} \tag{3}$$

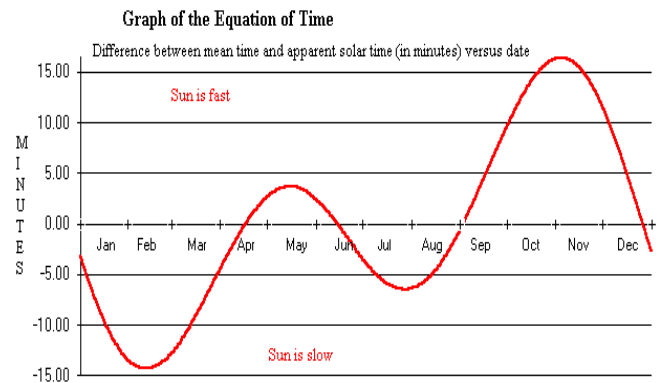The variation of EOT with respect to month in a year is shown below

Fig – 1- Equation of Time(minutes)

For the conversion between solar and clock time, we need knowledge of the location, N and the local standards. The conversion between t_s and LCT (local clock time) in 24hours is given below

$$\text{LCT} = t_s - \frac{\text{EOT}}{60} + \text{LC} + \text{D hours} \tag{4}$$

Where D is one if the location is in a region with daylight savings time other it is zero and EOT should be in minutes and LC is longitude correction and is given by

$$\text{LC} = \frac{(\text{local longitude}) - (\text{longitude of standard time zone meridian})}{15}$$

hours   (5)

Declination angle (δ) is defined as an angle between the lines drawn between the centre of the earth and sun, during noon $\delta = 23.5^0$ in summer (summer solstice), and $\delta = -23.5^0$ in winter (winter solstice).

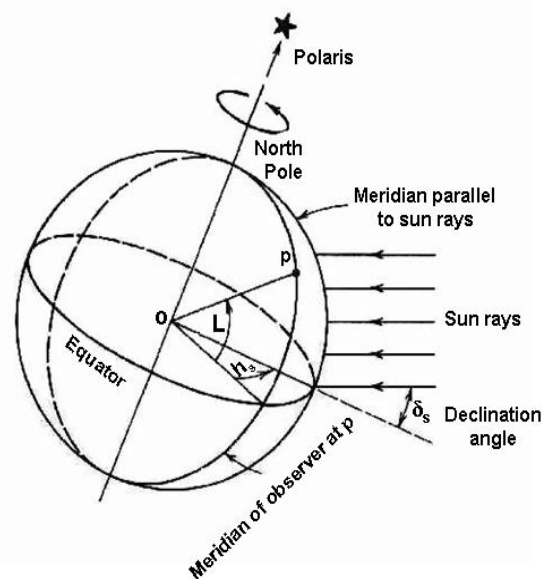$$\text{Sin } \delta = 0.93795 \cos [0.98563 (N - 173)] \tag{6}$$

Fig – 2 Earth-Sun Angles

The position of the sun relative to the coordinates on the surface of the earth is described by two angles which are solar altitude and the solar zenith angle. The all angles are described with respect to the central ray from the sun.

The solar altitude angle (α) is defined as the angle between the central ray from the sun, and the horizontal plane containing the observer and is given in terms of zenith angle (θ_z) which is the complement of α.

$$\theta_z = 90^o - \alpha \text{ degree} \tag{7}$$

$$\alpha = \sin^{-1}(\sin \delta * \sin \phi + \cos \delta * \cos \omega * \cos \phi) \text{ degrees} \tag{8}$$

where ϕ is latitude of the particular location

Azimuth angle (A) is an angle measured horizontally clockwise from north base line till the meridian at which observer is present and is given by

$$A' = \sin^{-1}(\frac{-\cos \delta \sin \omega}{\cos \alpha}) \qquad \text{degrees} \tag{9}$$

Where if $\cos \omega \geq ((\tan \delta)/(\tan \phi))$ then $A=180^o - A'$

 Where if $\cos \omega < ((\tan \delta)/(\tan \phi))$ then $A=360^o + A'$

Using above Earth sun equations (1-9) we can get the position of the sun with relative to the observer[12]. In this study dual axis tracker is used which tracks both altitude and azimuth angles.

2.2 MPPT:

A MPPT controller plays an important role in PV system. The duty of the MPPT is to increase the efficiency of the PV module by adjusting the output power of PV to the PV module by adjusting the output power of PV to its MPP under all conditions. There are so many and different topologies are available in the literature. MPPT technique is mainly applied to the system whose output

Power varies with time and is not constant for a time period.

The main problem faced by MPPT[13] is that the power transfer efficiency of the solar panel depends on two factors, one is the amount of sunlight falling and other is the electrical characteristics of load. As we know that sunlight varies with time, so the corresponding load characteristics which gives highest power transfer efficiency changes. So, in order to have high efficiency the load characteristics should also change w.r.t change in the sunlight, at this particular load characteristics we can say that MPP is achieved and the task of the MPPT to find MPP is also accomplished.

In this study P&O[14] algorithm is used. Which perturbs the PV module current and voltage and power is calculated, and it was compared with previous value to determine perturbation in next step.
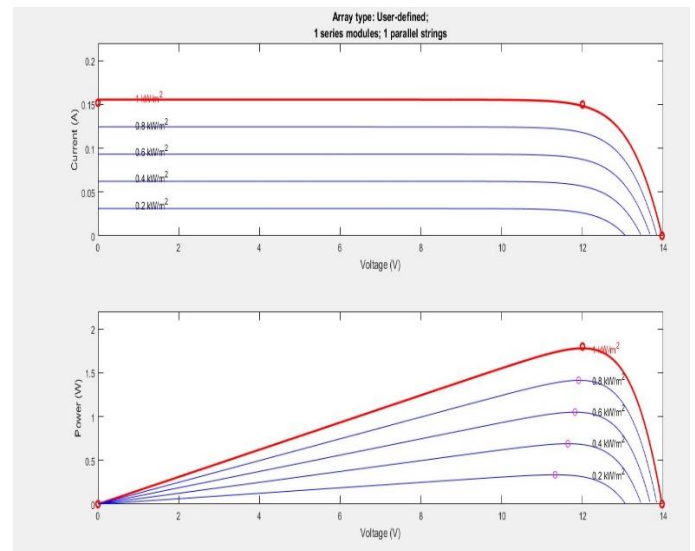

Fig – 3 V vs I and P vs V (MPP)

2.3 Node-RED:

There are three main types of nodes in node-red. First is input node allows us to give input to the node-red (e.g. inject), second is output node which allows us to send data to outside services like twitter, email, Etc, next is memory node or processing node which allows us to process data, these nodes have both input and one or more output. Using third type of nodes we can incorporate functionalities like delay, trigger, storing and receiving data from a file, mathematical operations, Etc.
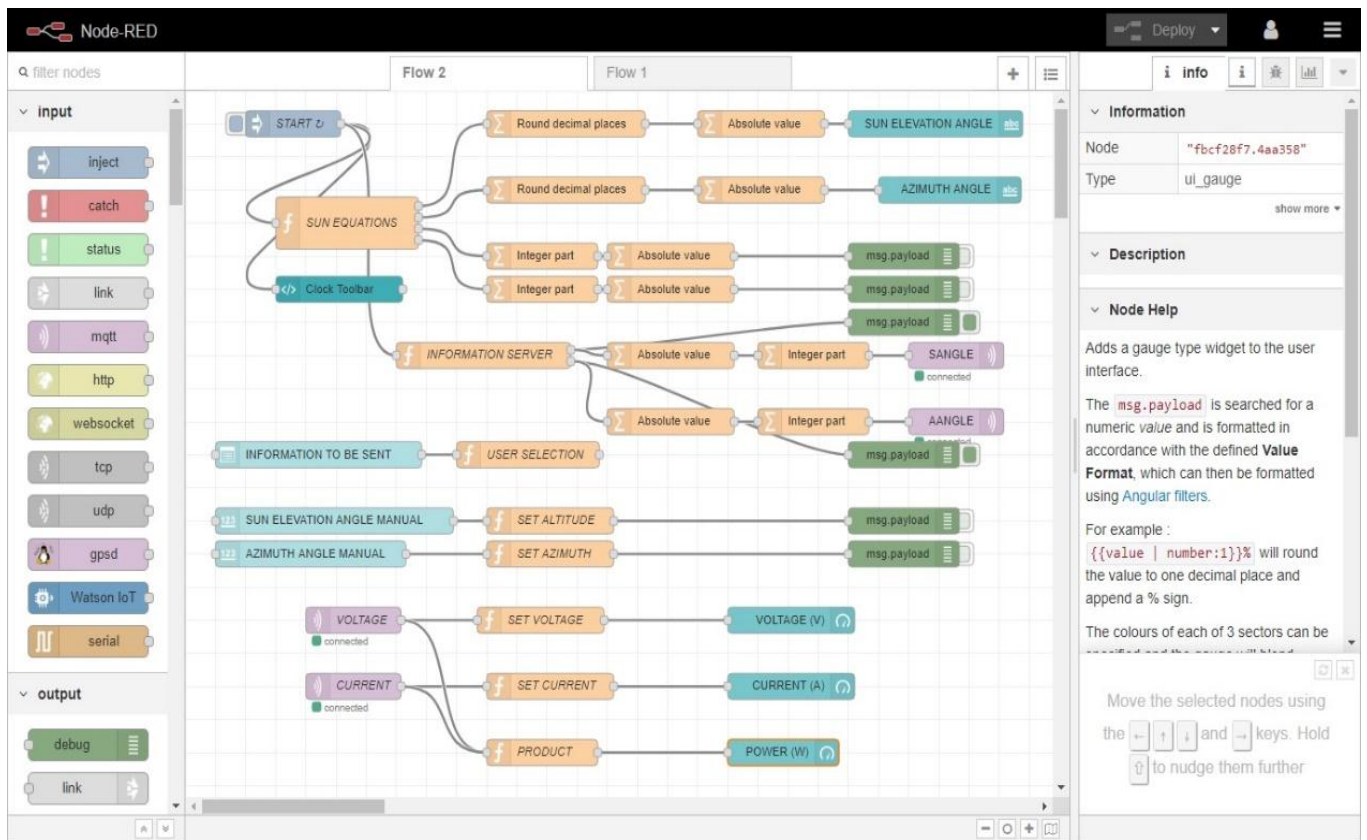
Fig – 4: Node-RED flow used in this study

Figure 4 showing the node palette (left), flow canvas (centre) and output pane (right) the main pane is the flow creation workspace in the middle. This is where we drag and drop nodes and connect them with wires. A set of tabs that can be seen on the top of the workspace. Previously created workspace can be accessed by the tabs, flows which are created using that previous workspace can also be seen. The output pane which is on the right-hand side of the screen is used to toggle between the info and debug tabs. When info is selected, information of the selected node is shown there. When debug is selected, it will display the output of debug nodes, errors and warnings.

For calculating the position of the sun, Earth-Sun equations algorithm is written in SUN EQUAITONS function node which takes the input as timestamp and gives us the azimuth and altitude angle, it will execute for every five seconds. Here we must provide our geographical location as an input to the sun equations to know the sun location.

The part of the program for sun equations is given below

Fig – 5 program for calculating sun angles

2.4 Electrical components:

1. **MG996R**[15] **servo:** Two identical servo motors are used for adjusting the both altitude and azimuth angles. This servo motor is having maximum stall torque of

11kg/cm, the angle of the servo motor can be easily controlled by connecting them to Node-MCU digital pins, It is less power consumption compared to stepper motor and cost is also less.

2. **Node-MCU**[16]: NodeMCU is used as a client for the server which can interact with hardware, and it has Wi-Fi MCU ESP8266 integrated in it which is best tool for IoT application development at the low cost.it has 16 GPIO pins, some which can be utilised for producing PWM signals up to 1MHz frequency.

3.**Boost converter:** boost converter[17] used in this system is shown in Figure – 6, it consists of one MOSFET(IRNZ44N) is used as switch, with a duty cycle of D=Ton/T, and constant switching time period of T=1/f, where f is the switching frequency and Ton is on time period. Here current through the load (I_L) is constant during steady state (PV voltage is constant) and is given as $I_L = V_{PV}/(R_{ds} + R_L)$, R_{ds} is drain source resistance. Current through the capacitor is ignored and the average load current is approximately equal to the PV output current.
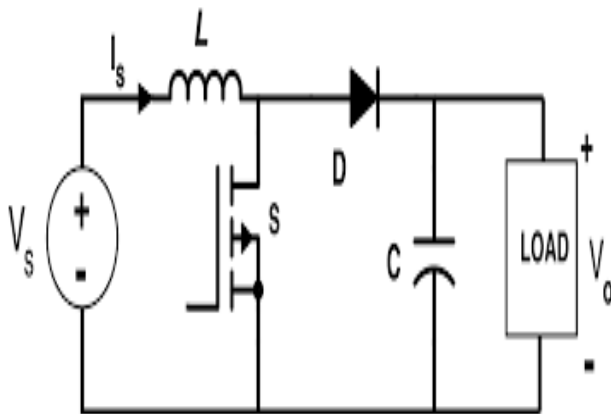
$$I_{PV} = D*I_L \qquad (10)$$

Fig – 6 Boost converter

4. *PV Module:* A commercially available PV module (12V 150mA) is used with following specifications.

Peak Voltage: 13.97V, Peak Current: 0.152A.

3. EXPERIMENTAL RESULTS

        A PV tracking system has been built and experimentally validate the theoritical results and performance of the proposed system. The MPPT logic is implemented in MPPT node in the Node-RED.
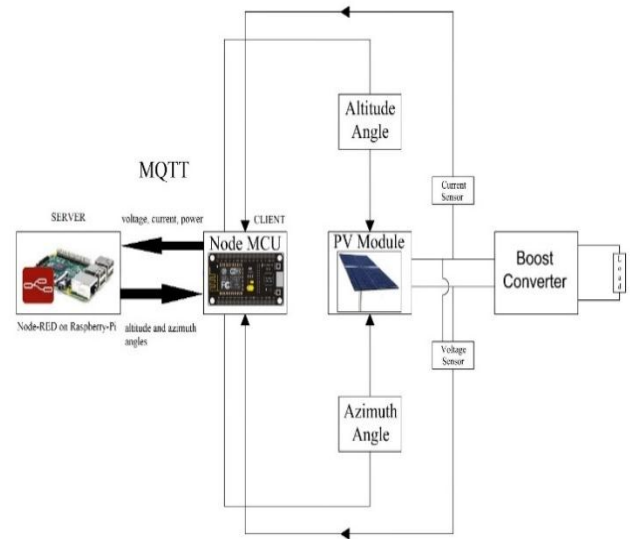

Fig – 7 Block diagram of the proposed tracker

NodeMCU is used as the controller for the PV tracker and the information for the NodeMCU is being sent from Node-RED server which is installed on raspberry pi through mosquitto broker using MQTT publish and subscribe protocol., here raspberry pi is a server and nodeMCU is a client. The MPPT logic is running on Node-RED and duty cycle information is sent to the nodeMCU and PWM pulses is given from the pwm pin of the MCU to the MOSFET in the Boost converter.

The overall blockdiagram of the proposed tracking system is shown in the Fig. 7 the PV output current is measured using ACS712 sensor and the voltage is measured using voltage sensor which is resistive based. Here load for the PV panel is resistive load. NodeMCU will send information of voltage, current, power of the PV panel to the Node-RED using MQTT protocol, and Node-RED will send Duty cycle, altitude and azimuth angle information to NodeMCU The information has to be converterd to the string data type for transmission between nodeMCU and Node-RED.

Two MG996R servo motors each for altitude and azimuth angle adjustments, when the nodeMCU receives the information of altitude and azimuth it will adjust he servos to the respective angles. These servos has an stall torque of 9.8kg/cm (4.8V). The node-red dashboard is designed to control and monitor the PV tracker remotely. To access the dashboard we have to type "[ipadress]:1880/ui" in the browser. The dashboard created in this study is shown in below fig.8. A dropdown selection is created on the dashboard which allows the user to send the which type of information, If it is selected as automatic node-RED will send altitude and azimuth angles of the sun which are calculated using sun equations. If we select as manual, on the right side down

of the panel we have to set the angles of which are send to nodeMCU and it adjusts these angles to the PV panel.
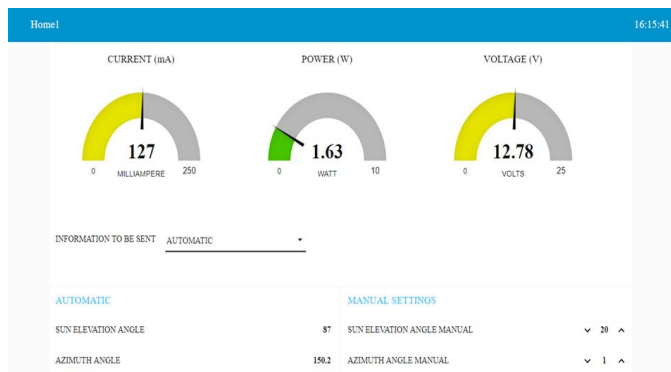

Fig – 8 Dashborad of the proposed tracker

In the experiment, the constructed sun tracker was activated. The actual data of altitude and azimuth angles of the sun position in the sky was obtained from the SUNCALC.ORG website at a geographical position of (17.6868° N, 83.2185° E) were measured from hour to hour in daylight (6:00-18:00) on Aug.10, 2019. Fig 10, Fig. 11 compare the actual altitude and azimuth angles of the sun position with the altitude and azimuth angles tracked by the tracker. The comparision shows that the average tracking error of altitude and azimuth angles are 0.1° and 0.47° respectively, which validates the theoretical result presented in section II. The overall experimental setup is shown in Fig.9 .
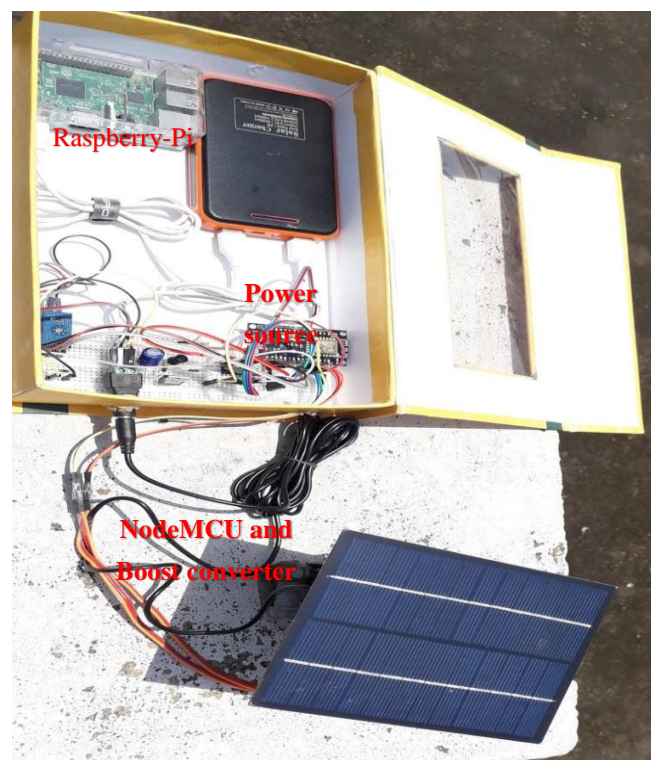

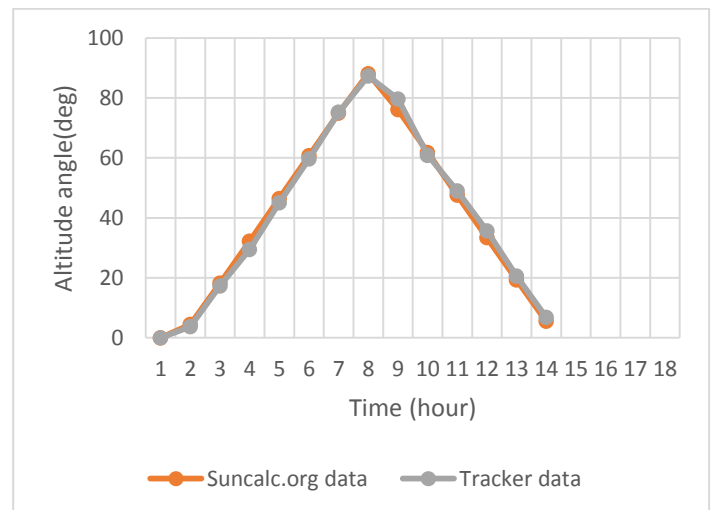**Fig – 9** The overall experimental setup

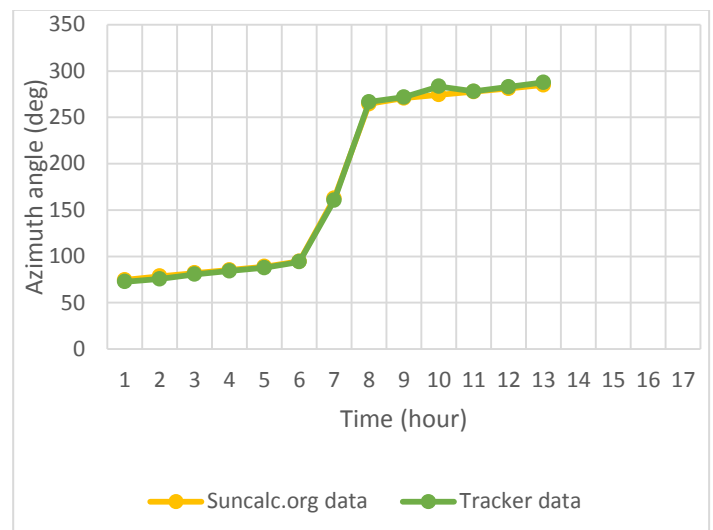**Fig – 10** comparision of actual and calculated altitude angles

Fig – 11 comparision of actual and calculated azimuth angles

The experiment was conducted in fixed mode and tracking mode, for fixed mode the angle of the pv is set to the value of the latitude angle. The fig.12 shows the comaprision of the Power ouput with both fixed and tracking mode. In fixed mode the total energy obtained during a day was 11.098 Wh and in tracking mode the total energy obtained was 14.64 Wh, so there is an increase in the energy efficiency of 31.92% using the proposed tracker.
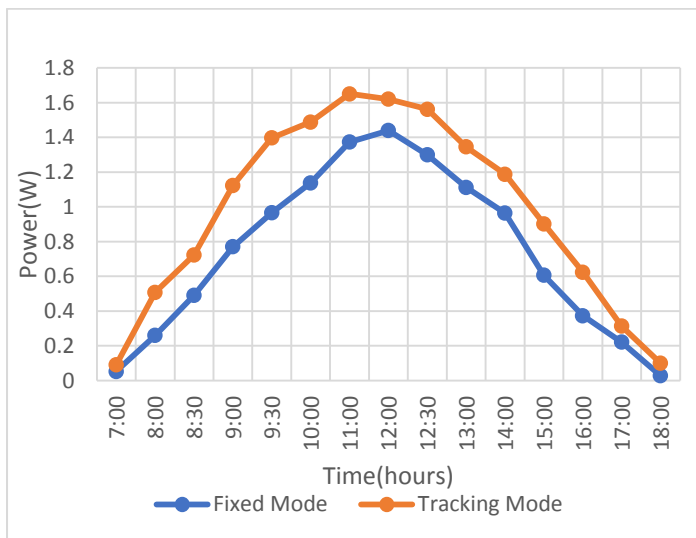
Fig – 12 comaprision of the Power ouput with both fixed and tracking mode

4. CONCLUSION

An IoT based sensor less dual axis sun tracker regulated by MPPT unit was proposed. This proposed system is very well used to test various tracking algorithms and to control and monitor the PV cell remotely which is very useful for research purpose. The advantage of the IoT enabled tracker is that it reduces the effort of collecting the data and analysing it, and to change the position of the PV module can be done remotely. The algorithm used in this study has successfully calculated the position of the sun with an average tracking error of both azimuth and altitude angles of only 0.285°.

REFERENCES

[1] L. Trappeniers, M. A. Feki, F. Kawsar, M. Boussard, "The Internet of Things: The next technological revolution", *Computer*, vol. 46, no. 2, pp. 0024-25, 2013.

[2] R. Mastromauro, M. Liserre, A. Dell' Aquila, "Control Issues in Single-Stage Photovoltaic Systems:MPPT Current and Voltage Control", *IEEE Trans. Industrial Informatics*, vol. 8, no. 2, pp. 241-254, May 2012.

[3]   July   2019   [online] https://www.electronicoscaldas.com/datasheet/MG996 R_Tower-Pro.pdf

[4]   July   2019   [online] https://www.nodemcu.com/index_en.html

[5] R. Gules, J. De, Pellegrin Pacheco, H. L. Hey, J. Imhoff, "A maximum power point tracking system with parallel connection for PV stand-alone applications", *IEEE Trans. Ind. Electron.*, vol. 55, no. 7, pp. 2674-2683, Jul. 2008.

[6] J.P. Conti, "The Internet of Things", *Proc. IEEE Commun. Eng.*, vol. 4, pp. 20-25, Dec./Jan. 2006.

[7] A. Sarma, J. Girao, "Identities in the future Internet of Things", *Wireless Pers. Commun.*, vol. 49, no. 3, pp. 353-363, May 2009.

[8] J. I. Vazquez, "Communication architectures and experiences for web-connected physical smart objects", *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, pp. 684-689, Mar. 2010.

[9] K. Ashton, "That "Internet of Things" Thing: In the Real World Things Matter More than Ideas", *RFID Journal*, May 2019, [online] Available: http://www.rfidjournal.com/articles/view?4986.

[10]   July   2019,   [online]   Available: http://www.nodered.org

[11] July 2019, [online] Available: http://Raspbian.org/.

[12]   July   2019,   [online]   Available: https://mosquitto.org/.

[13]   July   2019,   [online]   Available: https://pubsubclient.knolleary.net/.

[14] M. Iqbal, An introduction to solar radiation, Toronto: Academic Press, 1983.

[15] J. J. Michalsky, "The astronomical almanac's algorithm for approximate solar position (1950–2050)", *Solar Energy*, vol. 40, pp. 227-235, 1988.

[16] J. Meeus, Astronomical Algorithms, United States of America:Willmann- Bell Inc., 1991.

[17] T. Esram, P. L. Chapman, "Comparison of photovoltaic array maximum power point tracking techniques", *IEEE Trans. Energy Convers.*, vol. 22, no. 2, pp. 439-449, Jun. 2007.