PROJECT REPORT

On

Solar Plant Monitoring
And
Predictive Maintenance

By

ARAVIND KHUMAR A,

2015104006,

B.E. EEE 4th YEAR,

COLLEGE OF ENGINEERING GUINDY,

ANNA UNIVERSIY.

Table of Contents:

Summary:

Today, a large number of government buildings are investing in solar photovoltaics to meet their annual energy requirements by installing rooftop solar panels at their office locations. This helps them to reduce their energy bills besides providing a source of clean and sustainable energy. It is imperative for plant owners and AMC team to ensure that all the components of the solar plant are performing with maximum efficiency. This can be achieved by continuous health monitoring of the solar plant data and notify the Admin of the building and AMC team by Predictive Maintenance.

The prototype developed during the course of the internship is used to monitor the solar photovoltaics and keep track of their performance metrics. If at all any discrepancies are predicted in metrics collected, as said above the admin of the building is notified of the imminent fault and AMC team is alerted for maintenance.

Abstract:

This prototype is developed to collect **voltage, current, power, energy generated** both at the input and output terminals of the inverter. The data is stored in the memory registers of the inverter for specific interval of time and is overwritten periodically. The aim is to retrieve the data stored in its memory registers by communicating with inverter through the **RS232 or RS485** port given with the help of **MODBUS communication protocol**. This data collected by the hardware is converted into cloud understandable data and sent to Online Cloud Platforms, where the data is stored and analysed for discrepancies in performance metrics. This data sent to the cloud forms the heart of this solar photovoltaic predictive maintenance system. Cloud is equipped to do calculations that compares the obtained values with standard values and decides the working efficiency of the plant under observation. This prototype consists of 2 parts.

1. **Hardware Part –** A device used to collect data from inverter, process it and send to the online cloud platform.
2. **Software Part –** A polling, data conversion and database system that collects data, converts the data and stores the data locally within the prototype in case of internet failure.

This prototype is completely based on **Internet of Things (IoT)** technology that is used to send the data to cloud through internet and connect various other prototype all over the world.

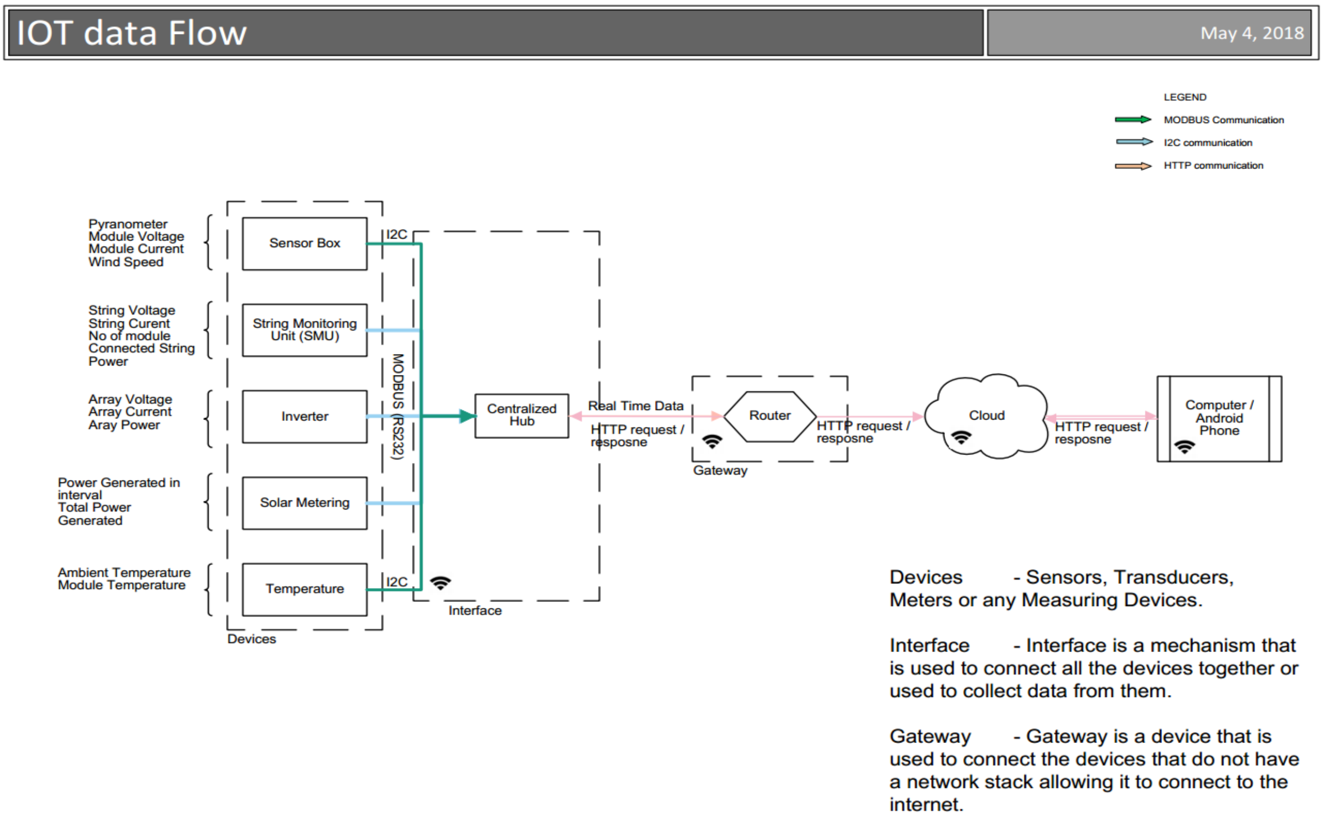
Scope:

The ability of the prototype to comprehensively record, analyse and assess data from all components of the solar plant helps to identify performance issues and their underlying causes. The system also provides insights about what may be causing the problem and suggest timely resolution to the problem and ensure optimum plant performance and also prevent energy loss for the owners of the rooftop PV solar plant.

Introduction:

The prototype for monitoring the PV solar plant was developed from scratch with the help of my mentor cum founder **Mr. Ashok Kumar** and online resources. The prototype was to be first installed in **rooftop** solar installations of **IES (Institute of Energy Sciences) in Anna University**. The capacity of the installations is **10KW** and was installed in the year **2015**. As an Internship trainee I was given knowledge about the different **types of solar plants**, their **architectures**, vital components and their working. Then I was taught about the basic principles involved in **Telemetry using Internet of Things**. After gaining deep knowledge about the **type of solar panels**, their vendors, ratings, type and vendor if the **Solar Grid Tied inverter (Growatt),** we started to develop the basic layout and data flow that would be required in the system. By visiting the official website of Growatt inverters we came to know that they have their own data-logger. But data collected through the proprietary data-logger wasn't very much customer friendly and didn't give insights about the maintenance metrics of the plant. So we decided to develop our own data-logger.

The flowchart given below gives a decent picture of how the planning went.

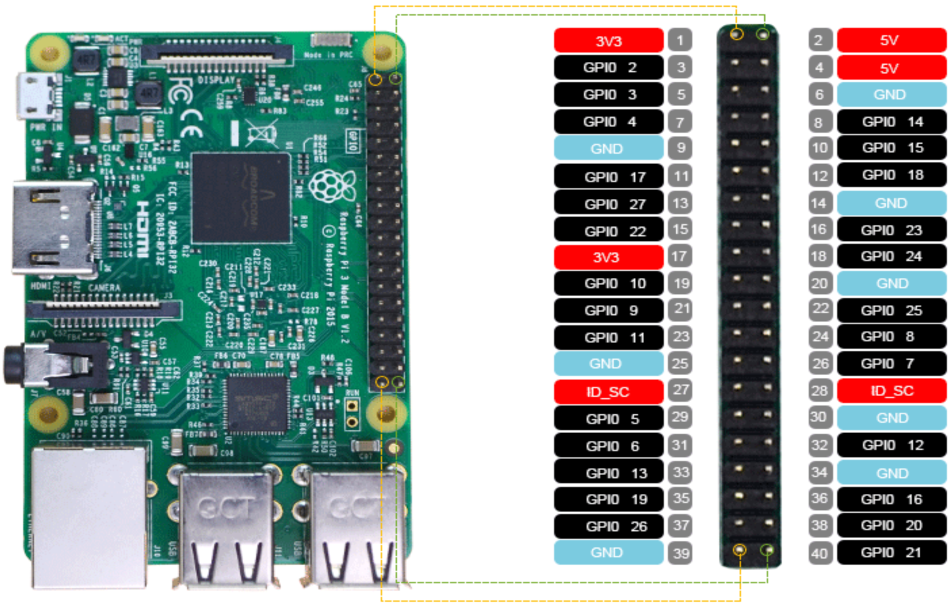


After analysing all possible hardware that could be used for achieving the above data flow we settled with the following components.

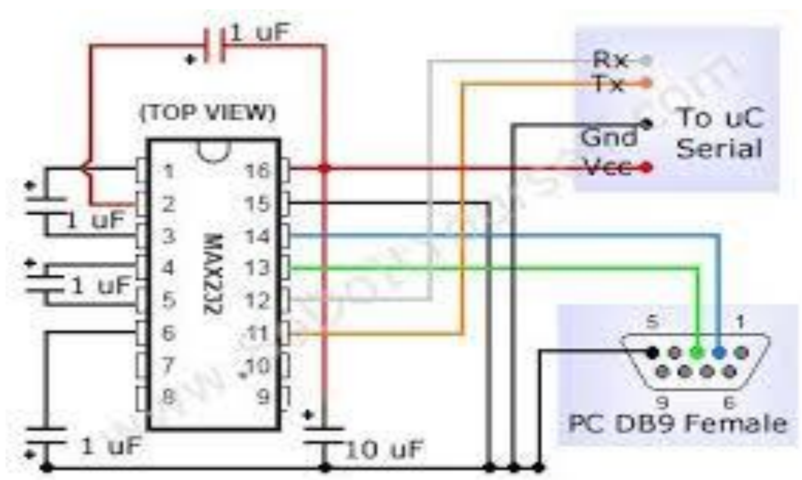
Components Used:

1. **Raspberry Pi 3** as Centralised Hub cum data-logger.
2. **RS232 to USB Converter** used to convert pinout of inverter to pin in of raspberry pi
3. **Wi-Fi Modem** as Internet Gateway.
4. **Power Source**

Raspberry Pi 3 is a single board computer that has WLAN and Bluetooth Connectivity. It runs on Linux Operating System. This acts as data logger system and collects data from the inverter by polling the inverter at periodical intervals using MODBUS protocol. It is programmed to do this logging function using Python 3.5 programming language. Also it converts the data obtained from the inverter into cloud understandable data and sends them to the cloud through HTTP protocol.



RS232 TO USB Converter is cable that acts an interface between the Growatt Inverter and the Raspberry Pi 3. This cable converts the data received from the RS232 port pins to Standard USB pinout type. Inside the cable there is an IC called MAX 232 that does the job of converting the voltage levels of the RS232 which is usually +3v to +15v for detecting High Level and -3v to -15v for detecting Low levels to USB type voltages which is +/- 5v.



Wi-Fi Modem is any router or data-card device that provide Wi-Fi hotspot so that the raspberry pi can connect to the internet and begin telemetry of the collected data.

Power Source provides 24/7 power to the prototype from 220V AC mains. A 5v, 1A adaptor is used here.

Technology Used:

Internet of Things (IoT) Telemetry:

Telemetry is gathering of data from remote places for analysis and other purposes. It is the heart of Industrial Internet of Things (IIoT). The spread of low powered wireless devices that could form themselves into mesh networks made this technology ubiquitous. It is possible to collect data from sensors that are placed remotely in human inaccessible places and monitor them through this technology. This data collected through telemetry are statefully stored in drives or clouds from analysis.

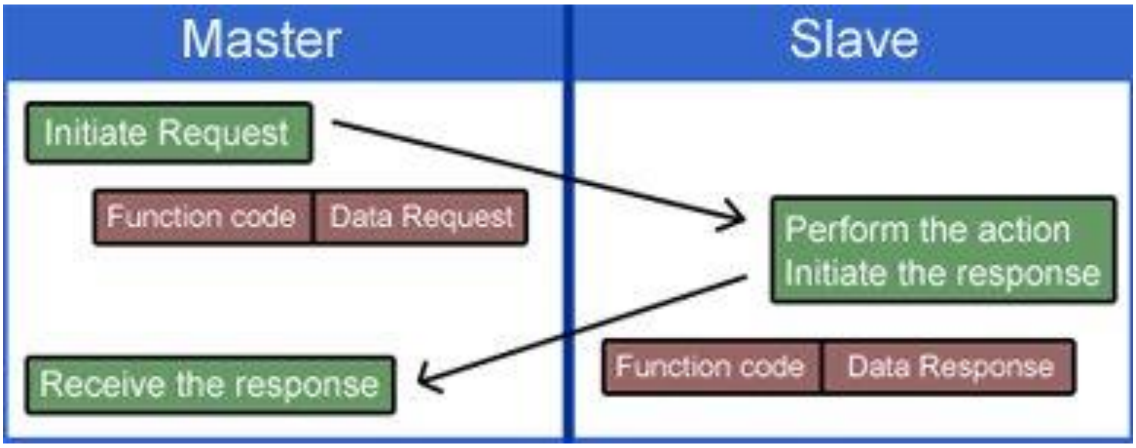
This telemetry is achieved through various protocols that makes it easier for identifying the channel or device type out of the millions of the device data it receives. Some of them are

HTTP, MQTT, CoAP, etc…

Protocols Used:

1. MODBUS:

What is Modbus?



Modbus is a serial communication protocol developed by Modicon published by Modicon in 1979 for use with its programmable logic controllers (PLCs). In simple terms, it is a method used for transmitting information over serial lines between electronic devices. The device requesting the information is called the Modbus Master and the devices supplying information are Modbus Slaves. In a standard Modbus network, there is one Master and up to 247 Slaves, each with a unique Slave Address from 1 to 247. The Master can also write information to the Slaves.

What is it used for?

Modbus is an open protocol, meaning that it's free for manufacturers to build into their equipment without having to pay royalties. It has become a standard communications protocol in industry, and is now the most commonly available means of connecting industrial electronic devices. It is used widely by many manufacturers throughout many industries. Modbus is typically used to transmit signals from instrumentation and control devices back to a main controller or data gathering system. Modbus is often used to connect a supervisory computer with a remote terminal unit (RTU) in C protocol exist for serial lines (Modbus RTU and Modbus ASCII) and for Ethernet (Modbus TCP).

2. HTTP (Hyper Text Transfer Protocol):

HTTP is the most popular and widely used protocol. It is document-centric and request-response protocol for client-server computing. HTTP protocol allows it to compose lengthy headers and messages.

Here in this prototype HTTP is used for telemetry because of the large size of its messages for telemetry.

SQLite3 – For Storing data locally:

SQLite is a C library that provides a lightweight disk-based database that doesn’t require a separate server process and allows accessing the database using a nonstandard variant of the SQL query language. Some applications can use SQLite for internal data storage

PyModBus – For Communication with Inverter:

Pymodbus is a full Modbus protocol implementation written in python. This library facilitate the use of python programming language for communication with devices that work with Modbus Communication protocols externally.

Programming Language for Raspberry Pi 3:

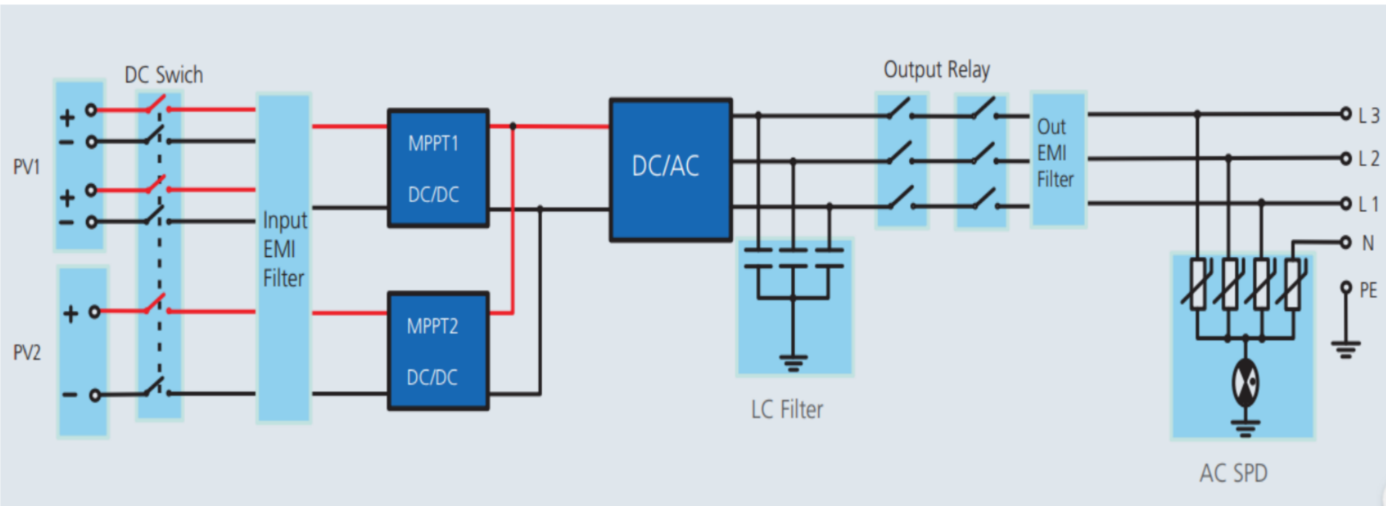
Python is an interpreted high-level programming language for general purpose programming. It provides constructs that enable clear programming on both small and large scales. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

UBIDOTS – IoT cloud platform:

Ubidots is an Internet of Things (IoT) data analytics and visualization company that provides a simple and secure connection for sending and retrieving data to and from cloud service in real-time. Its application enablement platform supports interactive, real-time data visualization (widgets). Ubidots exists as an easy and affordable means to integrate the power of the IoT into business or research.

Our Findings about the 10KW Plant at IES:

If the power generated by the PV Modules exceed more than 6.5 kW out of 10 kW of the installed capacity, then the MPPT efficiency of the Inverter reduces to less than 20%. This event happens during peak sun-hours generally from 10.00 am to 2.00 pm. The plant has been installed with Inverter of capacity 10kW with 2 MPPTs each connected with PV string of 5 kW.



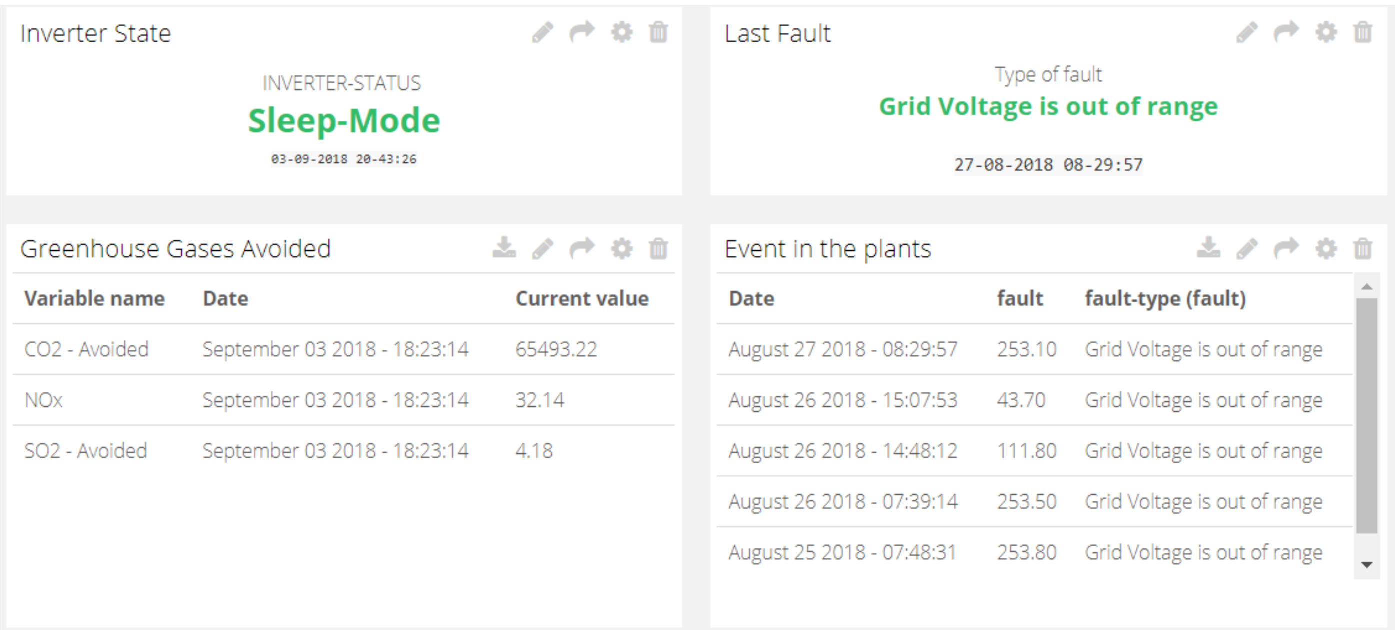
Peak Sun-hours = 10 am to 2 pm.
Average Input power generated during Peak Sun-hours = 7.5 kW
Peak Sun-hours duration = 4 hrs.
Total energy generated by PV modules = 30 kWh

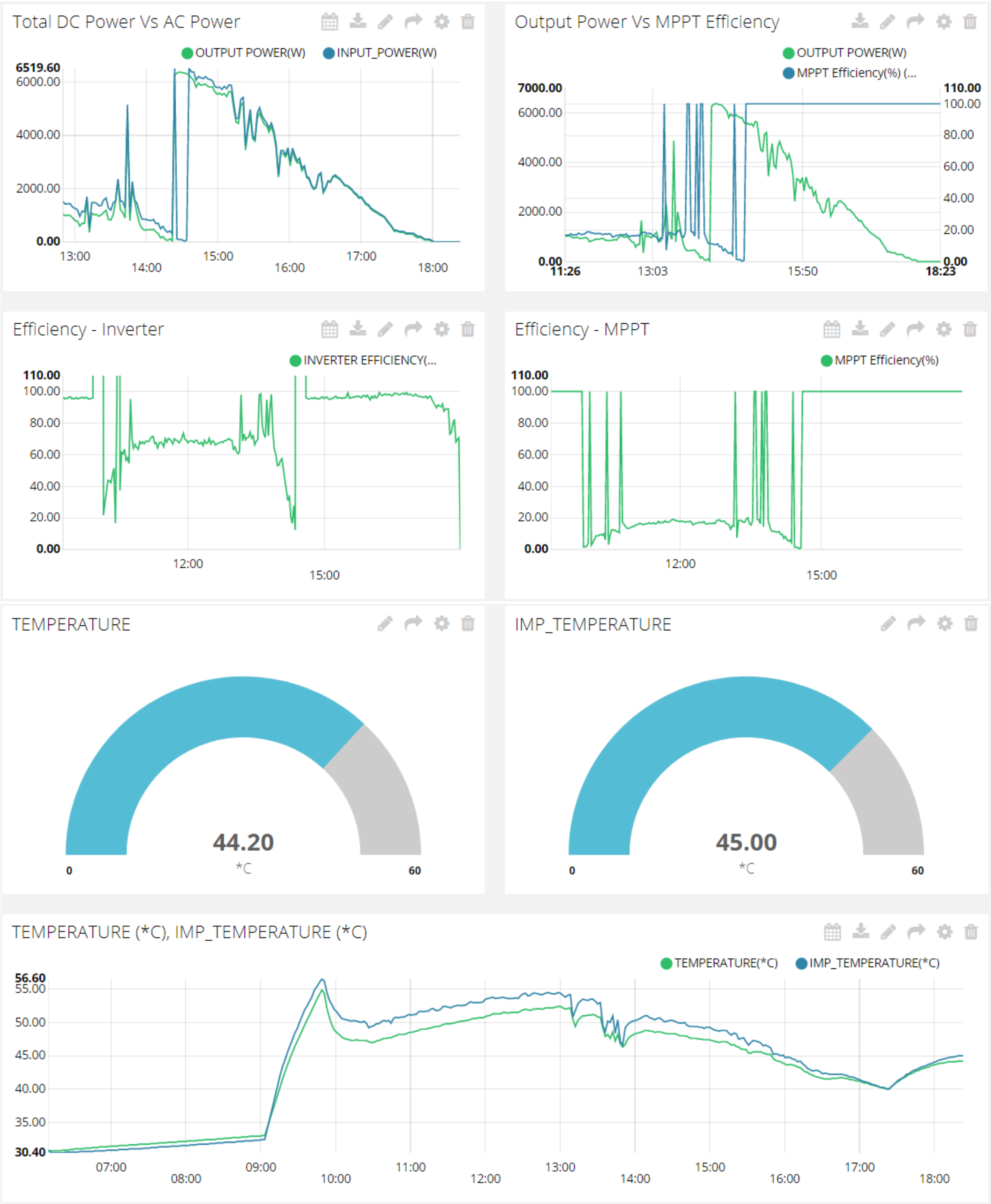
MPPT is operating with the efficiency less than 20% during the Peak sun-hours.

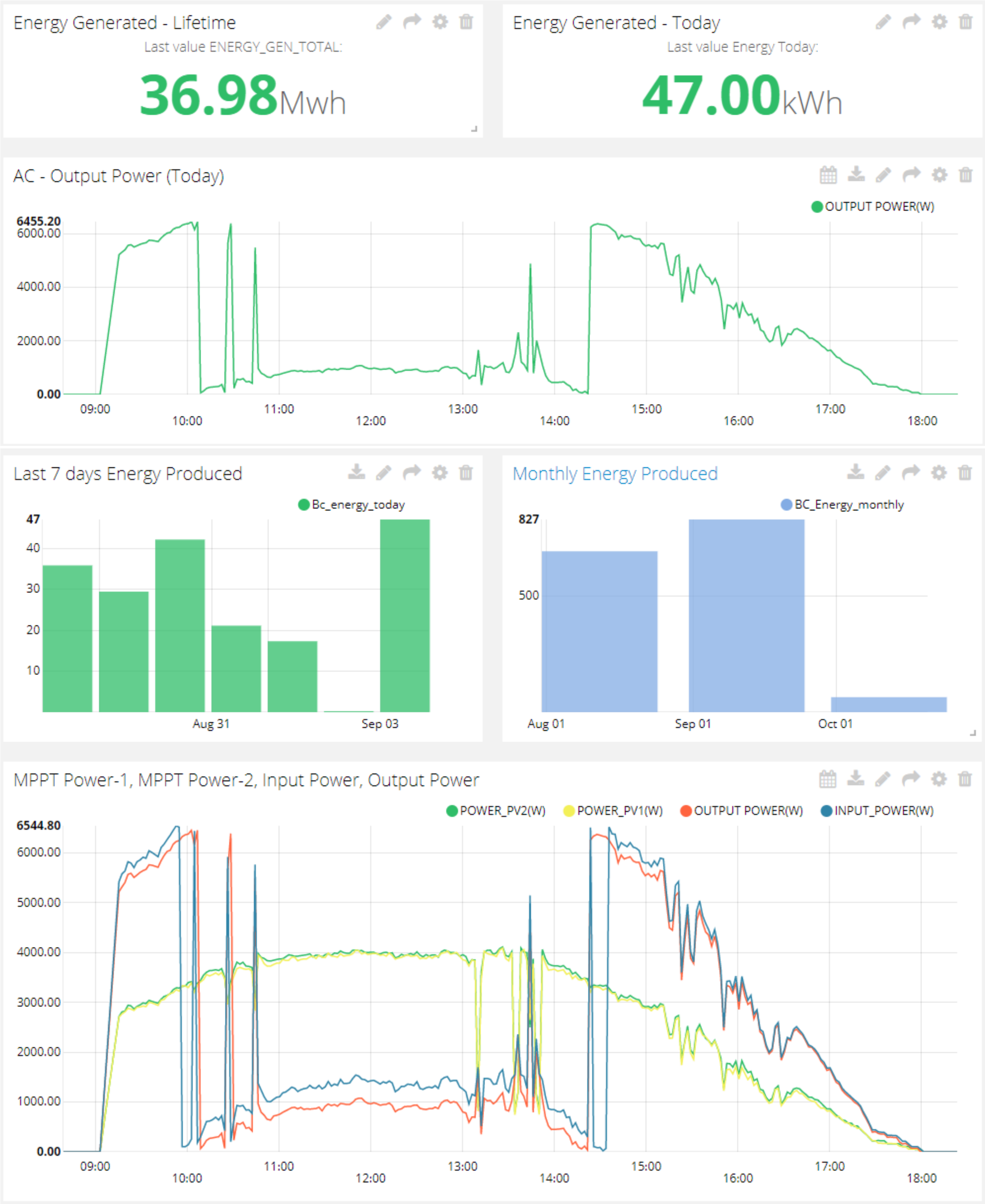
MPPT Loss = 80%
Total energy loss = Total energy generated by PV modules (kWh) X MPPT Loss (%)
Total energy loss = 24 kWh
The case study showing the impact of **loss of energy of around 240 kWh/ Month** if the above event is not rectified for a long time.

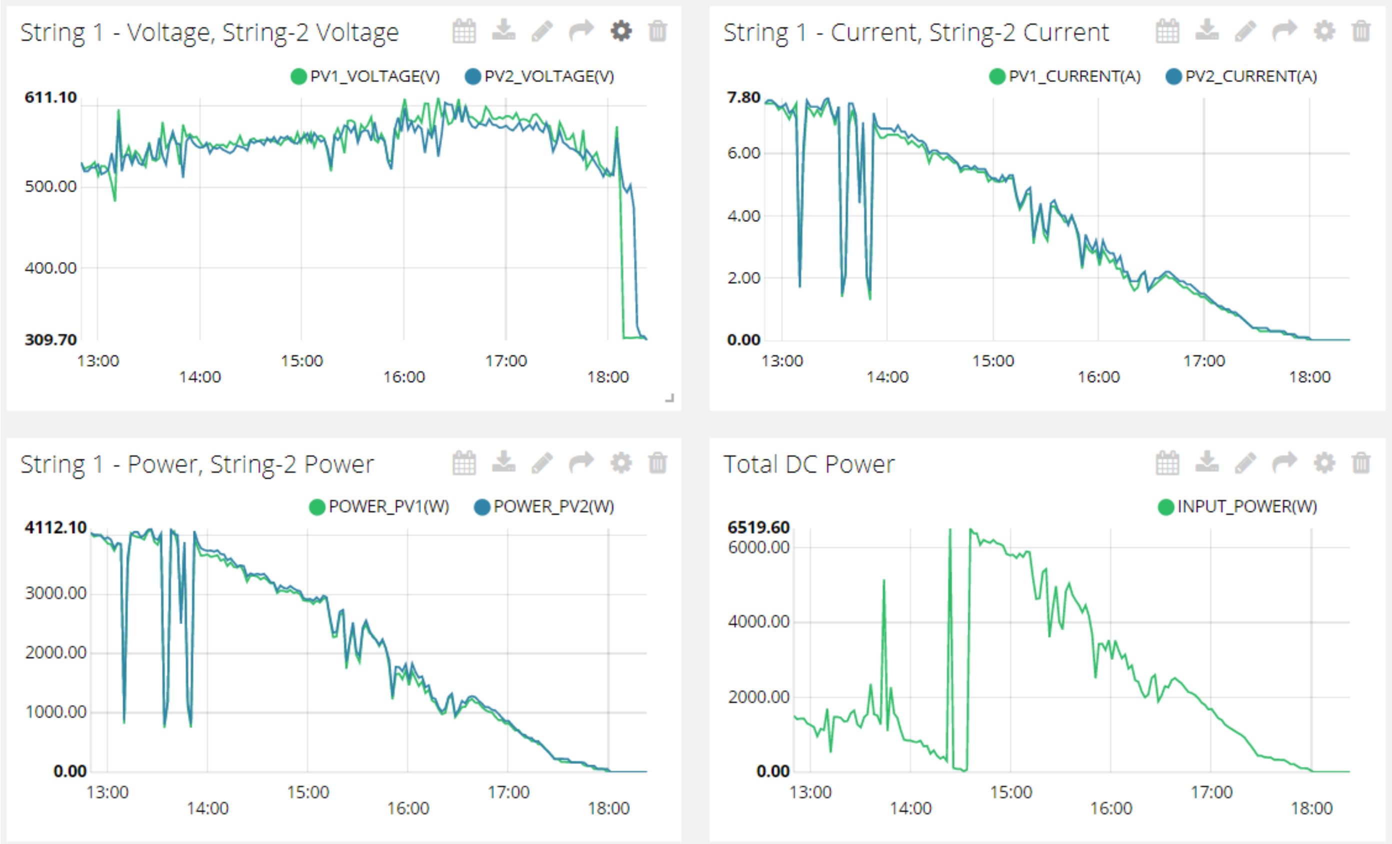
Energy loss per day in units = 24 kWh (Approximate)
No of day with energy loss in a month = 10 day
Total energy loss in units for a month = 240 kWh (Approximate)

<u>Performance Metrics Derived from data in Cloud:</u>

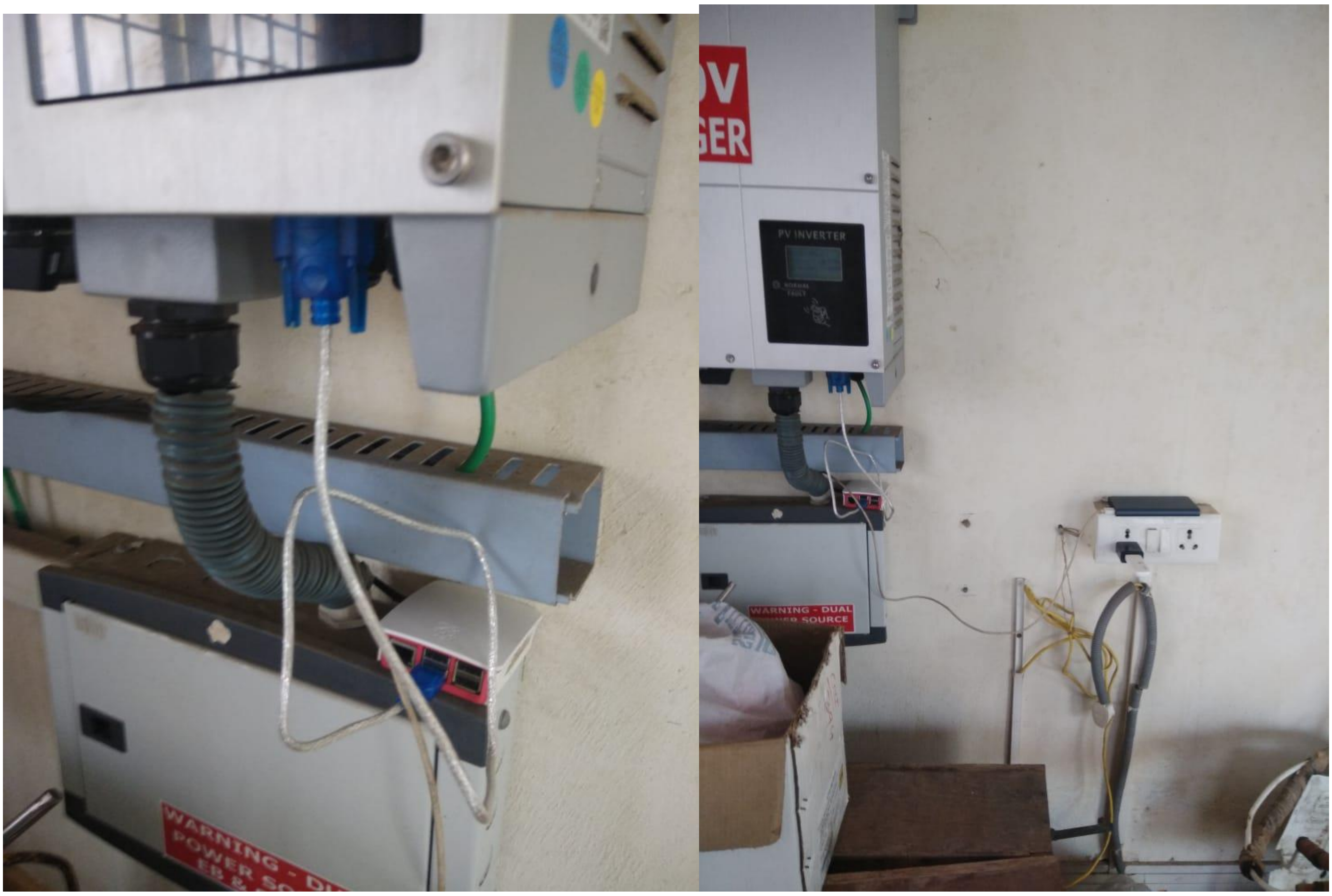



Total DC Power Vs AC Power
OUTPUT POWER(W)
INPUT_POWER(W)
6519.60
6000.00
4000.00
2000.00
0.00
13:00
14:00
15:00
16:00
17:00
18:00
Output Power Vs MPPT Efficiency
OUTPUT POWER(W)
MPPT Efficiency(%) (...
7000.00
6000.00
4000.00
2000.00
0.00
110.00
100.00
80.00
60.00
40.00
20.00
0.00
11:26
13:03
15:50
18:23
Efficiency - Inverter
INVERTER EFFICIENCY(...
110.00
100.00
80.00
60.00
40.00
20.00
0.00
12:00
15:00
Efficiency - MPPT
MPPT Efficiency(%)
110.00
100.00
80.00
60.00
40.00
20.00
0.00
12:00
15:00
TEMPERATURE
44.20
*C
0
60
IMP_TEMPERATURE
45.00
*C
0
60
TEMPERATURE (*C), IMP_TEMPERATURE (*C)
TEMPERATURE(*C)
IMP_TEMPERATURE(*C)
56.60
55.00
50.00
45.00
40.00
35.00
30.40
07:00
08:00
09:00
10:00
11:00
12:00
13:00
14:00
15:00
16:00
17:00
18:00

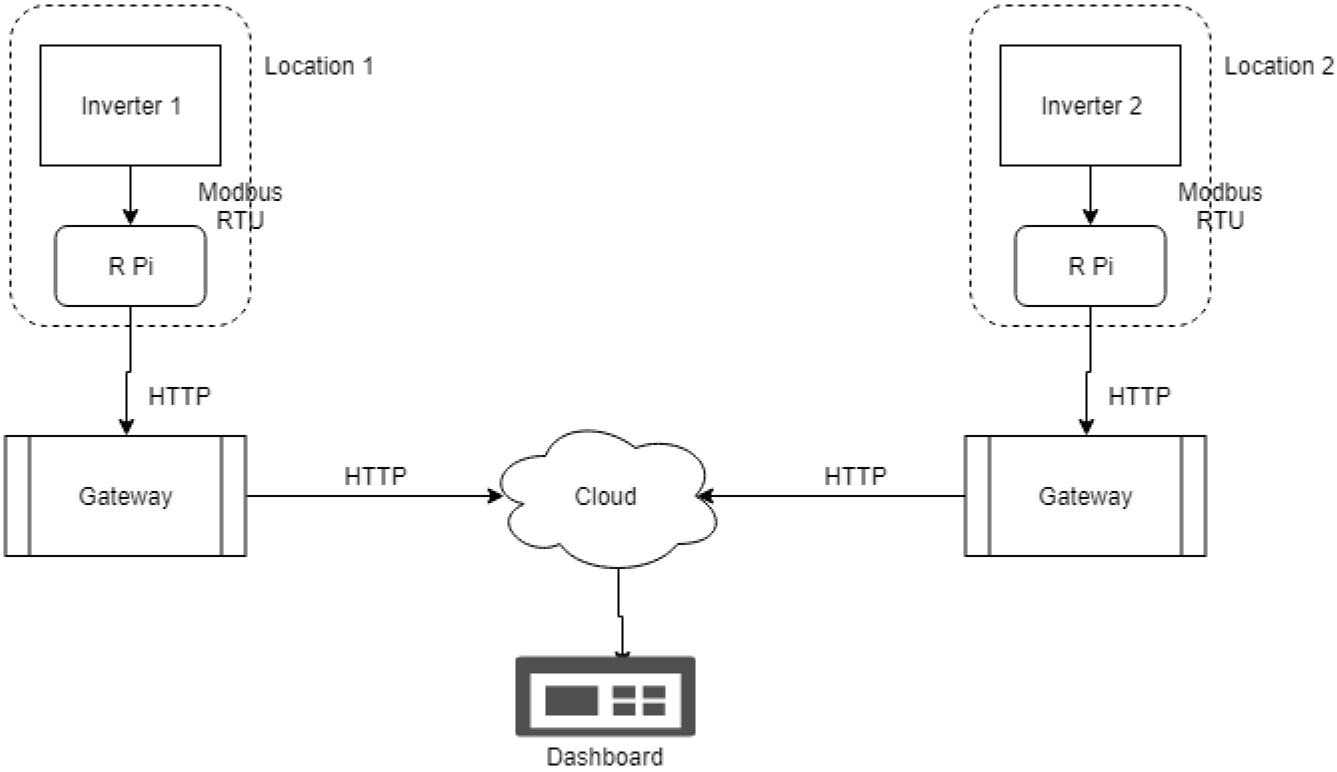

Energy Generated - Lifetime
Last value ENERGY_GEN_TOTAL:
36.98Mwh
Energy Generated - Today
Last value Energy Today:
47.00kWh
AC - Output Power (Today)
OUTPUT POWER(W)
6455.20
6000.00
4000.00
2000.00
0.00
09:00
10:00
11:00
12:00
13:00
14:00
15:00
16:00
17:00
18:00
Last 7 days Energy Produced
Bc_energy_today
47
40
30
20
10
Aug 31
Sep 03
Monthly Energy Produced
BC_Energy_monthly
827
500
Aug 01
Sep 01
Oct 01
MPPT Power-1, MPPT Power-2, Input Power, Output Power
POWER_PV2(W)
POWER_PV1(W)
OUTPUT POWER(W)
INPUT_POWER(W)
6544.80
6000.00
5000.00
4000.00
3000.00
2000.00
1000.00
0.00
09:00
10:00
11:00
12:00
13:00
14:00
15:00
16:00
17:00
18:00



<u>**Finalised Design of the Prototype:**</u>



Minimalistic Flow Chart of the Deployed Prototype:



Conclusion:

The prototype is under post developmental stages and needs minor bug fixes according to the needs. Thus a simple Single board computer is used as a data logging device cum telemetry IoT device when connected to the internet. These low power devices have the potential to the change the way we perceive ideas about automation to a greater scale. This prototype will be further developed by Artro Energy as their product. Machine learning algorithms can be incorporated to turn the decision making process more accurate and insightful. This product when completed will act as a complete hardware that helps a solar plant investor to monitor his plant and ROI be as minimum as possible.

References:

1. SQLite official site - https://www.sqlite.org/index.html
2. Python official site - https://www.python.org
3. Internet of Things Wikipedia - https://en.wikipedia.org/wiki/Internet_of_things
4. Ubidots IoT platform – https://www.ubidots.com
5. Raspberry Pi - https://www.raspberrypi.org/
6. Pymodbus - https://pypi.org/project/pymodbus/
7. Telemetry - https://cloud.google.com/solutions/iot-overview
8. Growatt Inverters - http://ginverter.com/
9. Using Raspberry pi with growatt - http://fixedbycode.blogspot.com/2014/04/how-to-upload-data-from-growatt.html