

Open Source SCADA Systems for Small Renewable Power Generation

**M.Eng Student: Lawrence O. Aghenta
Thesis Supervisor: Prof. M. Tariq Iqbal**

November 7, 2019

Overview:

1. Introduction
2. SCADA
3. Design and Dynamic Simulation of a Hybrid Power System
4. Open Source SCADA 1 Design: using Emoncms, Arduino Uno, Raspberry Pi, Ethernet & Node-Red
5. Open Source SCADA 2 Design: using Thinger.IO, ESP32 Thing & Wi-Fi
6. Open Source SCADA 3 Design: using ESP32 with OLED, ThingsBoard, Wi-Fi & MQTT Protocol.
7. Conclusions
8. Key Contributions
9. Future Recommendations
10. Acknowledgements
11. Q/A & References

Open Source SCADA Systems for Small Renewable Power Generation

Introduction:

Motivation:

1. The distributed nature of Power Generation infrastructures
2. High cost of Commercial SCADA systems: Economically unjustifiable for smaller applications.
3. Interoperability issues, high power consumption, & the need for expensive standard communication systems in

Commercial SCADA solutions.
4. The need for a cost-effective, reliable, secure, timely, flexible & sophisticated coordinated data monitoring and control

system: Open Source SCADA

Introduction:

Objectives, Challenges and Work Around:



a. Objectives:

- To design, model & dynamically simulate a Small Hybrid Power System.
- To study some proven Commercial SCADA Systems & Open Source SCADA packages.
- To study and test the available open source Internet of Things (IoT) platforms & use the best options.
- To design some low-cost, low-power, reliable, secure IoT-based Open Source SCADA systems.
- To test the designed Open Source SCADA systems with a Small Renewable Power Generation system with ESS.

b. Challenges:

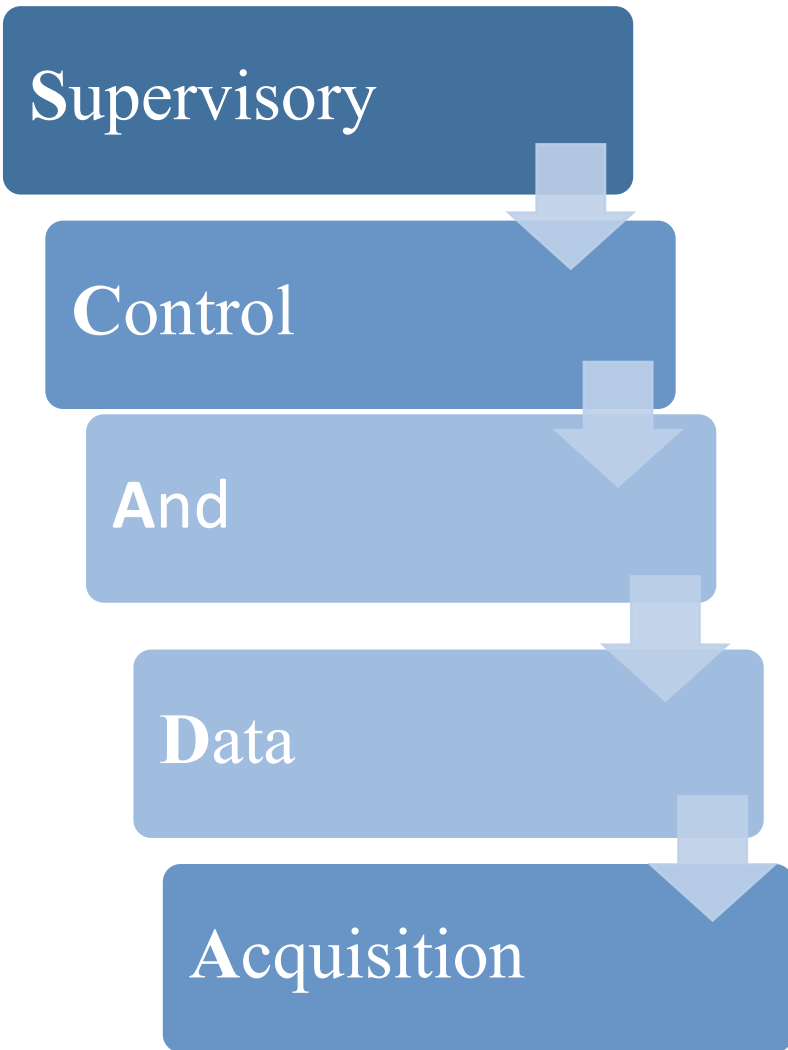
- Site for the Hybrid Power System Design.
- Choice of SCADA components to meet low-power objective.
- Security Strategies to meet security objective.
- Complexity of the entire system

c. Work Around:

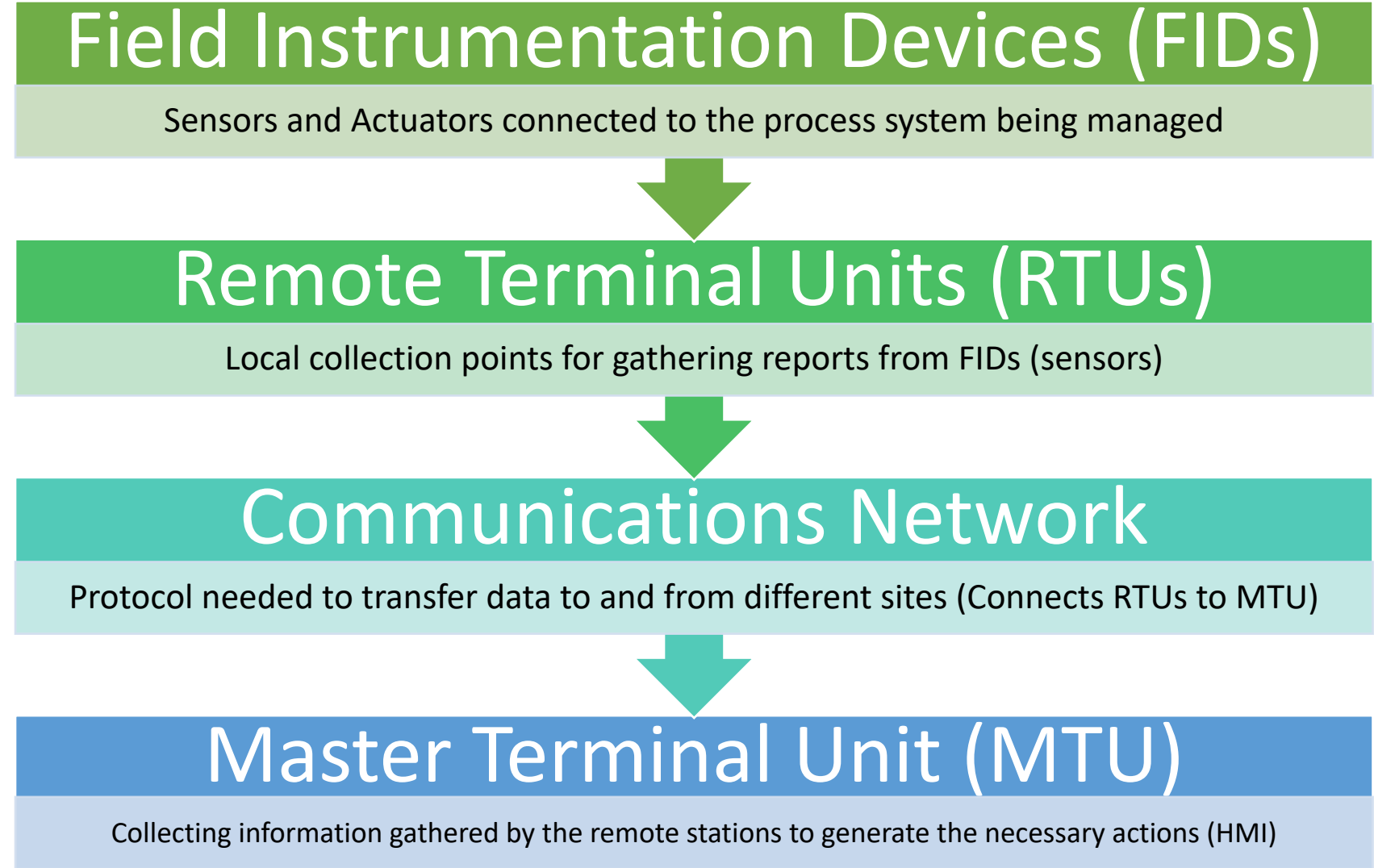
- A House in Nigeria was chosen for the Hybrid Power System Design.
- Tested Multiple components, and IoT Platforms.
- The main data server was locally installed, self-managed and self-hosted on own private network.
- Simple solutions proposed to overcome the complexity of the entire system.

SCADA:

SCADA: Description & Basic Elements



SCADA: Basic Elements



SCADA:

SCADA: Basic Functions

Data Acquisition

Networked Data
Communication

Data Presentation

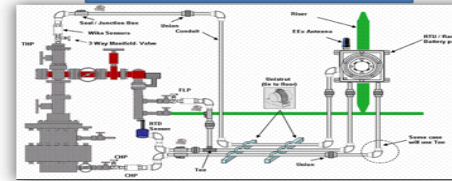
Monitoring/Control

SCADA Applications

Electric power generation,
transmission and distribution



Oil and Gas Production



Buildings, facilities and
environments



Mass transit



Water and sewage



SCADA: Architectures (Generations)

Monolithic

- 1st Generation

Distributed

- 2nd Generation

Networked

- 3rd Generation

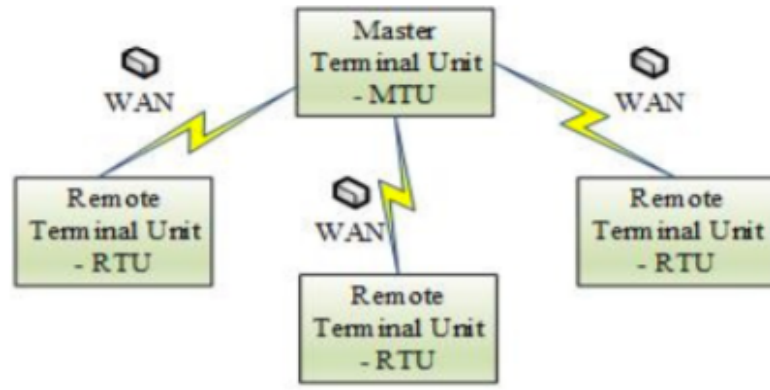
Internet of Things

- 4th Generation

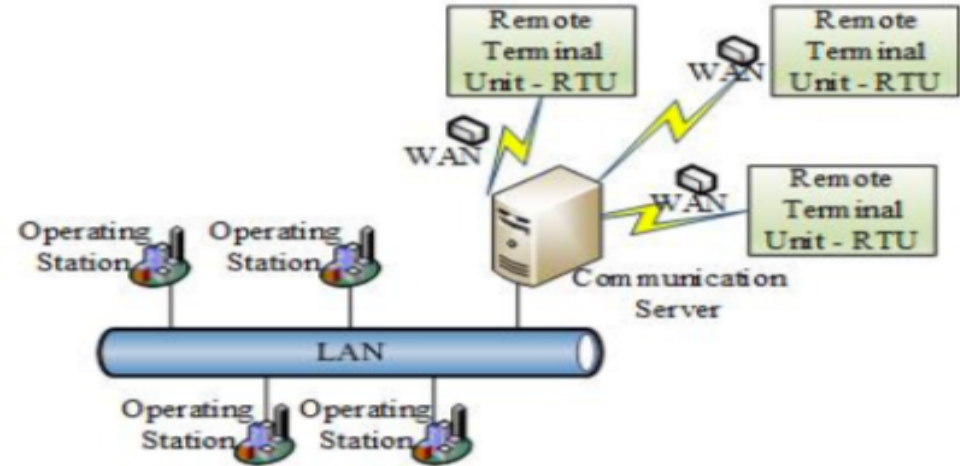
SCADA:

SCADA: Architectures (Generations)

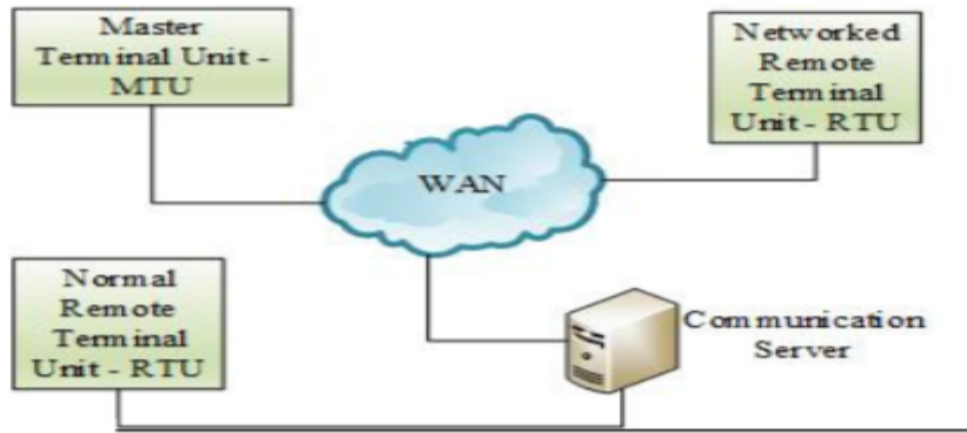
1st Generation - Monolithic SCADA:



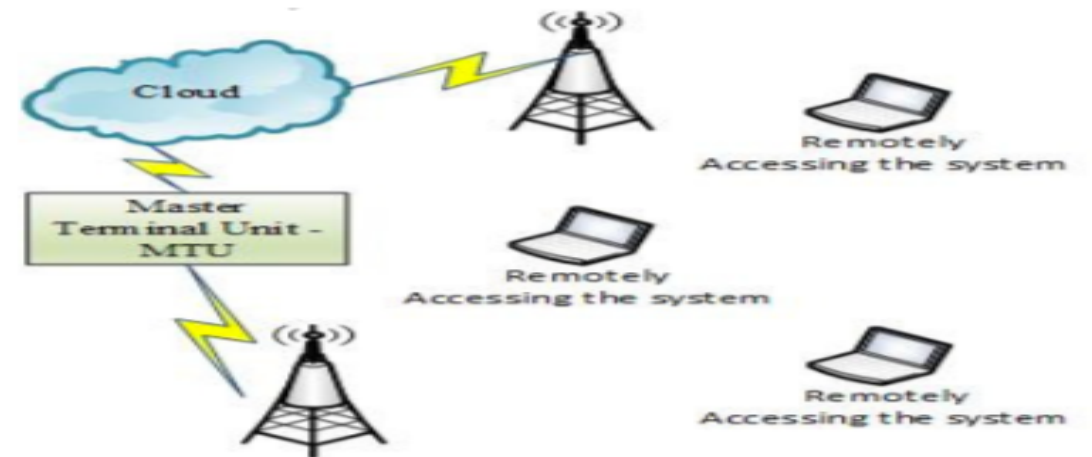
2nd Generation - Distributed SCADA:



3rd Generation - Networked SCADA:



4th Generation - Internet of Things (IoT)-based SCADA:



SCADA:

Desired Characteristics:

- Dynamism; flexibility.
- Retrofit; upgrading/updating
- Ease of Use/Ease of Installation
- Reliability and Availability
- Low Power Consumption
- Security: various attacks are possible; Security techniques must be decided and implemented.

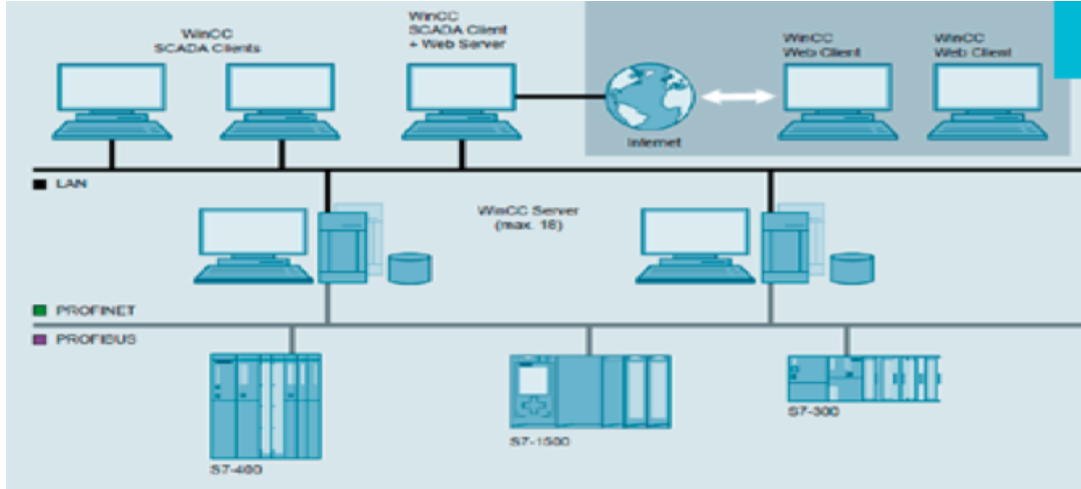
Classes of SCADA:

- Proprietary (Commercial) SCADA; e.g. Siemens, ABB, GE, Wonderware, Schneider Electric, etc.
- Open Source SCADA

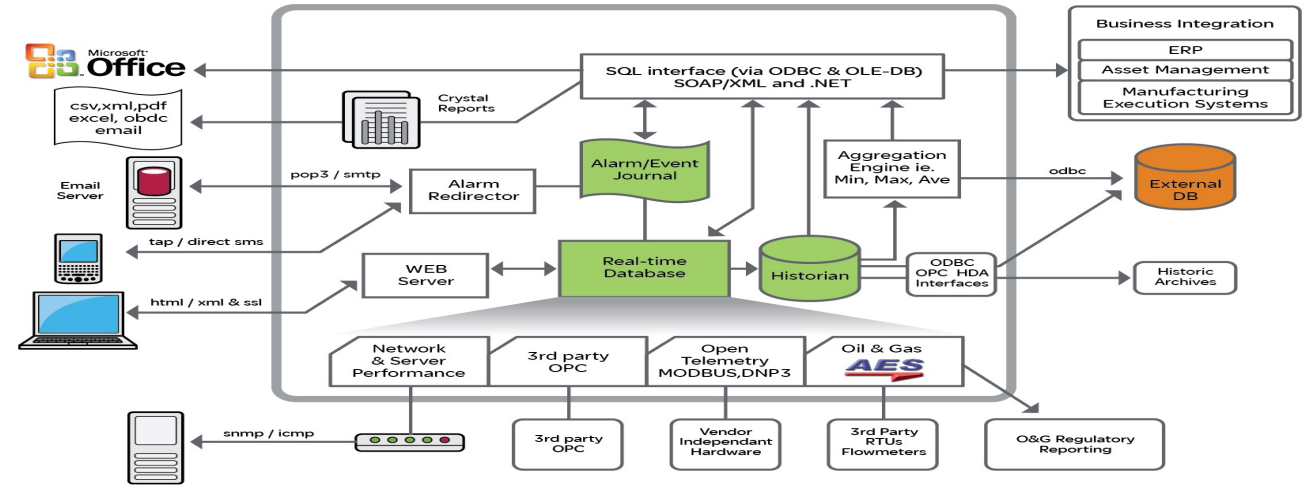
SCADA:

Some Studied Commercial SCADA Solutions:

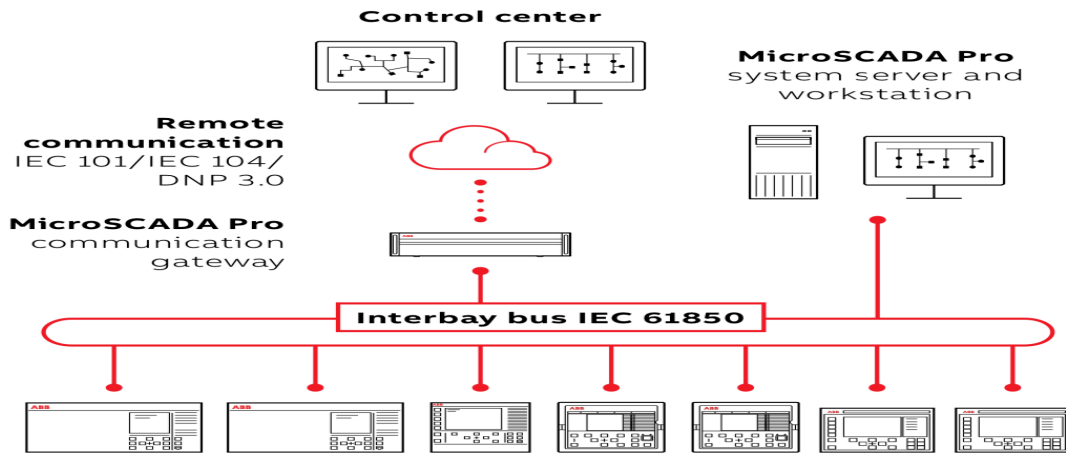
1. SIMATIC WinCC: Siemens



2. ClearSCADA: Schneider Electric



3. MicroSCADA Pro: Allen Bradley



4. Ovation SCADA: Emerson

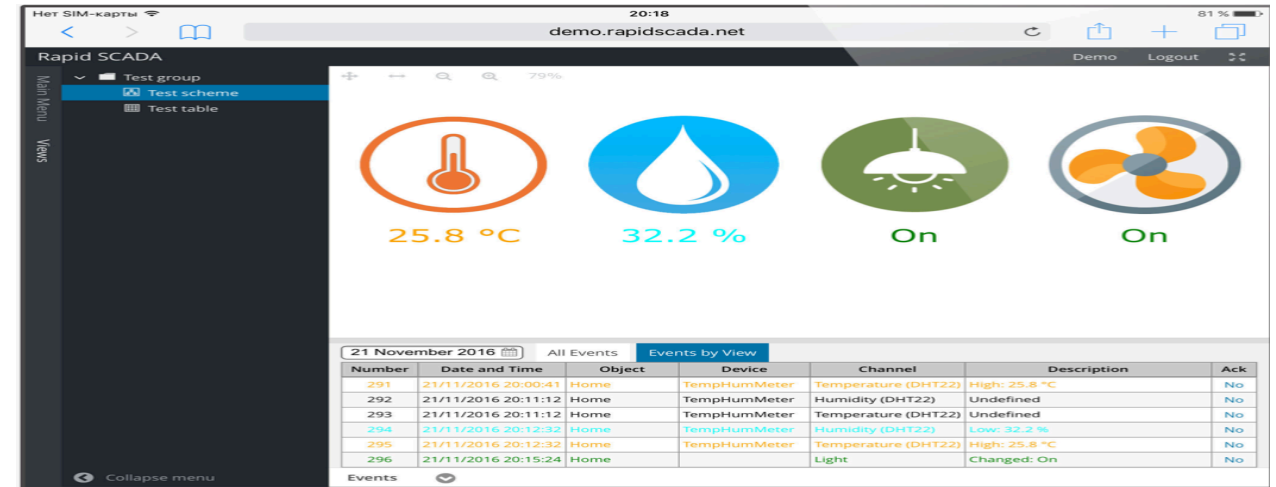
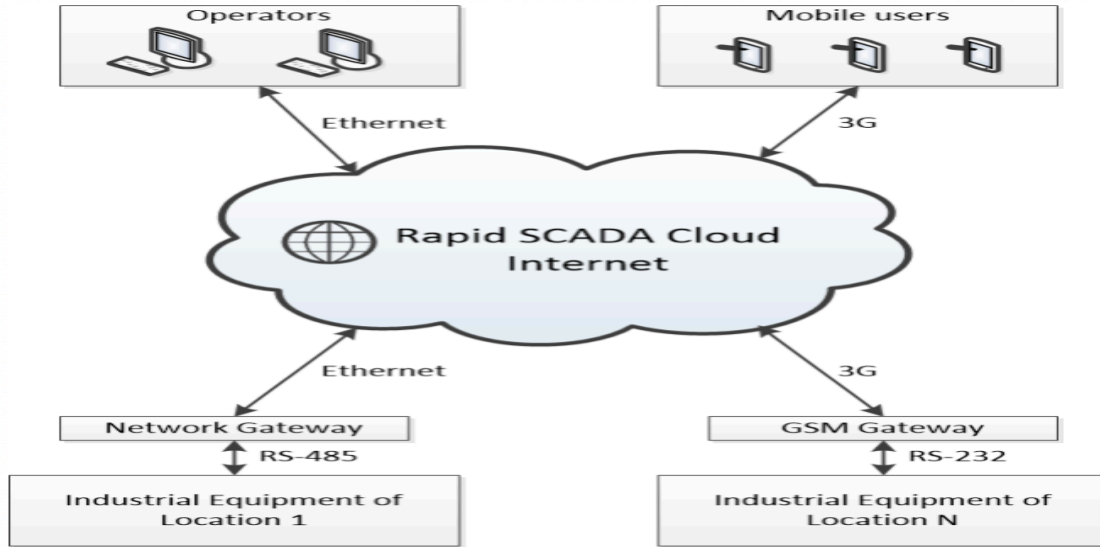
Specifications

SCADA Communication Server Specifications	
Hardware Platform	Windows
Standard Communication Protocols	Modbus (Master/Slave), Allen-Bradley DF-1, Allen-Bradley EIP, DNP3 (Client/Server), IEC61850, IEC60870-101, IEC60870-104, GE GSM, Turbine Control Interface, OPC UA (Client/Server)
Communication Media	Leased lines, dial-up telephones, microwave, licensed and unlicensed radio, spread spectrum packet radio, public switched networks, satellites, cellular, Ethernet (LAN and WLAN)
Redundancy Scheme	Primary and backup; supports dual redundant configuration
Number of Simultaneous Communication Channels	Up to 32 per server

SCADA:

Some Studied Open Source SCADA Solutions:

1. Rapid SCADA



2. Tango SCADA



Motivation:

- Extreme Electricity shortage and prolonged periods of power outages in Nigeria.
- Abundant Solar Irradiation in Nigeria (an average of $17.3 \text{ MJ/m}^2/\text{day}$ ($4.81 \text{ KWh/m}^2/\text{day}$) in Benin City).
- Modelling and Simulation is the necessary first step in design, optimization and performance analysis.
- A Case Study of a Hybrid Power System with a Small Renewable Power Generation.

Project Steps:

- House Energy Needs determination: Thermal Modelling and Analyses using BEOpt software
- System Sizing, Optimum Design Technology and Cost Analyses using HOMER Pro software
- Dynamic/Mathematical model of each of the subsystems (components).
- Matlab/Simulink Simulation of the Proposed PV System Component of the Hybrid Power System
- Simulation results.

Design and Simulation of a Hybrid Power System for a House in Nigeria:

House Energy Needs Determination: Thermal Modelling and Analysis (BEOpt):



Site Screen & House Picture:

BEOpt 2.8.0.0 - My Nigerian House Simulation 2* [Standard, New Construction, Single-Family Detached]

File Screen Case Run Reports Tools Graphs Help

Input: Output:

My Case My Case (2) My Case (3)

Analysis: Design Reference: My Design

This case contains output associated with these inputs and therefore inputs are disabled. To modify inputs or create a new design, either [clear](#) the existing output or create a [new case](#).

Building

EPW Location: GHA_Accra.654720_SWERA.epw

Terrain: City

Natural Gas Hookup:

Economics

Project Analysis Period: 30 years

Inflation Rate: 2.4 %

Discount Rate (Real): 3.0 %

Efficiency Material Cost Multiplier: 1.000

Efficiency Labor Cost Multiplier: 1.000

PV Material Cost Multiplier: 1.000

PV Labor Cost Multiplier: 1.000

Electricity Natural Gas Oil Propane

Utility Rates

Simple Detailed

User Specified Fixed: 8.00 \$/month

State Average Marginal: 0.1180 \$/kWh

National Average Average: 0.1267 \$/kWh

Fuel Escalation (Real): 0.00 %/year

Mortgage

Down Payment: 0.0 %

Mortgage Interest Rate: 4.0 %

Mortgage Period: 30 years

Marginal Income Tax Rate, Federal: 28.0 %

Marginal Income Tax Rate, State: 0.0 %

Other

Incentives: PV

Efficiency (Whole-Building)

Demand Response: Signals

Project Info

Building Name: My House in Nigeria

Street Address: Jonathan Akpobare Street, BDPA

City: Benin

State: Edo

Zip:

Country: Nigeria

Notes:

Please note that there's no weather data for Nigeria in BEOpt, but the weather for Nigeria is similar to that of Ghana as they are both West African Countries, and Accra is quite close to Lagos, Nigeria, which in turn is close to Benin City. Therefore, I have used the weather data for Accra, Ghana for this simulation.

PV Compensation

Net Metering Feed-in Tariff

Annual Excess Sellback Rate

Retail Electricity Cost: 0.03000 \$/kWh

User Specified

Monthly Grid Connection Fee: 0.00 \$/kW

Energy Factors

Source/Site Ratio: 3.150

Carbon Factor: 1.530 lb/kWh



Side View of the Chosen House

Design and Simulation of a Hybrid Power System for a House in Nigeria:



House Energy Needs Determination: Thermal Modelling and Analysis (BEOpt):

Main Floor and Building Foundation:

BEopt 2.8.0.0 - My Nigerian House Simulation 2* [Standard, New Construction, Single-Family Detached]

File Screen Case Run Reports Tools Graphs Help

Input: [Home] [List] [Globe] Output: [Bar Chart]

My Case My Case (2) My Case (3)

Analysis: Design Reference: My Design

This case contains output associated with these inputs and therefore inputs are disabled. To modify inputs or create a new design, either [clear](#) the existing output or create a [new case](#).

Levels: Fnd 1st 2nd 3rd 4th 5th 6th

Spaces: Living Garage Erase

Living: 2375 sqft

Wall Height: 8 ft

	Beds	Baths	Finished (sqft)
Total	5+	3+	2375

No errors.

Scale: 1 cell = 2 ft

Front

BEopt 2.8.0.0 - My Nigerian House Simulation 2* [Standard, New Construction, Single-Family Detached]

File Screen Case Run Reports Tools Graphs Help

Input: [Home] [List] [Globe] Output: [Bar Chart]

My Case My Case (2) My Case (3)

Analysis: Design Reference: My Design

This case contains output associated with these inputs and therefore inputs are disabled. To modify inputs or create a new design, either [clear](#) the existing output or create a [new case](#).

Levels: Fnd 1st 2nd 3rd 4th 5th 6th

Spaces: Slab Crawlpace Unfinished Basement Finished Basement Pier & Beam Erase

Slab: 2375 sqft

	Beds	Baths	Finished (sqft)
Total	5+	3+	2375

No errors.

Scale: 1 cell = 2 ft

Front

Design and Simulation of a Hybrid Power System for a House in Nigeria:



House Energy Needs Determination: Thermal Modelling and Analysis (BEOpt):

Sample Modelling Outputs:

House Energy Needs:

BEOpt 2.8.0.0 - My Nigerian House Simulation 2* [Standard, New Construction, Single-Family Detached]

File Screen Case Run Reports Tools Graphs Help

Input: [Icons] Output: [Icons]

My Case My Case (2) My Case (3)

Analysis: Design Reference: My Design

This case contains output associated with these inputs and therefore inputs are disabled. To modify inputs or create a new design, either [clear](#) the existing output or create a [new case](#).

My Design

- Orientation: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
- Neighbors: 1 2 3 4 5 6 7 8 9
- Walls
 - Wood Stud: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
 - Double Wood Stud: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
 - Steel Stud: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
 - CMU: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
 - SIP: 1 2 3 4 5 6 7 8 9
 - ICF: 1 2 3 4
 - Other: 1 2 3 4
 - Wall Sheathing: 1 2 3 4 5 6 7 8 9 10 11 12
 - Exterior Finish: 1 2 3 4 5 6 7 8 9 10 11
- Ceilings/Roofs
 - Unfinished Attic: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 2
 - Roof Material: 1 2 3 4 5 6 7 8 9 10 11 12 13
 - Radiant Barrier: 1 2
- Foundation/Floors
 - Slab: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
 - Carpet: 1 2 3 4 5 6
- Thermal Mass
 - Exterior Wall Mass: 1 2 3 4 5 6 7
 - Partition Wall Mass: 1 2 3 4 5 6 7
 - Ceiling Mass: 1 2 3 4 5 6 7
- Windows & Doors
 - Window Areas: 1 2 3 4 5 6 7 8 9 10 11
 - Windows: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
 - Interior Shading: 1 2 3 4 5 6 7
 - Door Area: 1 2 3 4

Option	Azimuth [degrees]
1) North	180.0
2) NNE	202.5
3) Northeast	225.0
4) ENE	247.5
5) East	270.0
6) ESE	292.5
7) Southeast	315.0
8) SSE	337.5
9) South	0.0
10) SSW	22.5
11) Southwest	45.0
12) WSW	67.5
13) West	90.0
14) WNW	112.5
15) Northwest	135.0
16) NNW	157.5

Orientation is defined as the direction faced by the front of the house (azimuth = degrees, clockwise from South).

BEOpt 2.8.0.0 - My Nigerian House Simulation 2 [Standard, New Construction, Single-Family Detached]

File Screen Case Run Reports Tools Graphs Help

Input: [Icons] Output: [Icons]

My Case My Case (2) My Case (3)

Select: My Design Tools: [Icons]

Category	Option	Cost (\$)	Description
Building	Orientation	\$0	South
Neighbors	Neighbors	\$0	Left/Right at 15ft, Front/Back
Walls	CMU	\$8,214	6-in Concrete Filled
Walls	Wall Sheathing	\$0	None
Walls	Exterior Finish	\$36,180	Brick, Medium/Dark
Ceilings/Roofs	Unfinished Attic	\$627	Ceiling R-7 Cellulose, Vented
Ceilings/Roofs	Roof Material	\$10,340	Galvanized Steel
Ceilings/Roofs	Radiant Barrier	\$0	None
Foundation/Floors	Slab	\$0	Uninsulated
Foundation/Floors	Carpet	\$0	0% Carpet
Thermal Mass	Exterior Wall Mass	\$841	1/2 in. Drywall
Thermal Mass	Partition Wall Mass	\$908	1/2 in. Drywall
Thermal Mass	Ceiling Mass	\$908	1/2 in. Drywall
Windows & Doors	Window Areas	\$0	18% F20 B40 L20 R20
Windows & Doors	Windows	\$7,531	Low-E, Double, Non-metal, Ai
Windows & Doors	Interior Shading	\$0	Apr-Sep = 0.5, Oct-Mar = 0.95
Windows & Doors	Door Area	\$0	20 ft^2
Windows & Doors	Doors	\$210	Wood
Windows & Doors	Eaves	\$1,970	2 ft
Windows & Doors	Overhangs	\$2,868	2ft, First Story, All Windows
Airflow	Space Conditioning		
Airflow	Space Conditioning Schedules		
		\$105,531	Total Present Value

Site Electricity Use (kWh/yr) breakdown for My Design:

- PV (E): 7222 kWh/yr
- Misc. (E): 1468 kWh/yr
- Lights (E): 6307 kWh/yr
- Hot Water (E): 1876 kWh/yr
- Cooling (E): 10868 kWh/yr

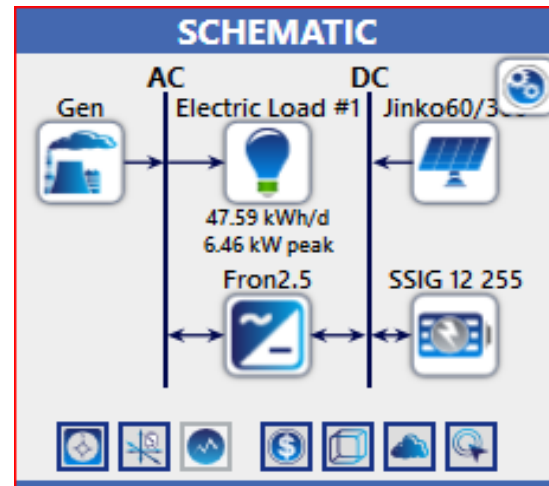
Previous runtime: 2m 6s

X: 0.00 (%/yr), Y: 1507.93 (\$/yr)

Design and Simulation of a Hybrid Power System for a House in Nigeria:

System Sizing, Optimum Technology and Cost: HOMER Pro:

- The generated BEOpt Annual Hourly Load Data exported into HOMER Pro for HPS Sizing, and optimum system design.
- HOMER (Hybrid Optimization of Multiple Energy Resources) software was developed by the National Renewable Energy Laboratory.
- HOMER models micropower systems with single or multiple power sources: e.g. Photovoltaics, Wind turbines, etc.
- HOMER, the micropower optimization model, helps to design off-grid and grid-connected systems.
- HOMER analyses help to answer design questions e.g: Components size? Most cost effective technology? etc.
- HOMER simulates various configurations to find the least cost combinations that meet electrical and thermal loads.



Hybrid Power System Components & Layout

Design and Simulation of a Hybrid Power System for a House in Nigeria:

System Sizing, Optimum Technology and Cost: HOMER Pro:



HOMER Simulation Results:

Simulation Results

System Architecture: Trojan SSIG 12 255 (6.00 strings)
 Jinko Eagle PERC60 300W (14.6 kW) Fronius Galvo 2.5-1 (3.36 kW)
 Autosize Genset (7.20 kW) HOMER Load Following

Total NPC: \$106,307.90
 Levelized COE: \$0.4734
 Operating Cost: \$5,650.04

Fronius Galvo 2.5-1 Emissions

Cost Summary Cash Flow Compare Economics **Electrical** Fuel Summary Autosize Genset Renewable Penetration Trojan SSIG 12 255 Jinko Eagle PERC60 300W

Production	kWh/yr	%
Jinko Eagle PERC60 300W	21,062	89.1
Autosize Genset	2,585	10.9
Total	23,647	100

Consumption	kWh/yr	%
AC Primary Load	17,372	100
DC Primary Load	0	0
Deferrable Load	0	0
Total	17,372	100

Quantity	kWh/yr	%
Excess Electricity	3,533	14.9
Unmet Electric Load	0	0
Capacity Shortage	0	0

Quantity	Value
Renewable Fraction	85.1
Max. Renew. Penetration	801

Monthly Average Electric Production

Report Copy Time Series: Plot... Scatter Plot... Delta Plot... Table... Export...

Simulation Results

System Architecture: Trojan SSIG 12 255 (6.00 strings)
 Jinko Eagle PERC60 300W (14.6 kW) Fronius Galvo 2.5-1 (3.36 kW)
 Autosize Genset (7.20 kW) HOMER Load Following

Total NPC: \$106,307.90
 Levelized COE: \$0.4734
 Operating Cost: \$5,650.04

Fronius Galvo 2.5-1 Emissions

Cost Summary Cash Flow Compare Economics **Electrical** Fuel Summary Autosize Genset Renewable Penetration Trojan SSIG 12 255 Jinko Eagle PERC60 300W

Quantity	Value	Units
Rated Capacity	14.6	kW
Mean Output	2.40	kW
Mean Output	57.7	kWh/d
Capacity Factor	16.4	%
Total Production	21,062	kWh/yr

Quantity	Value	Units
Minimum Output	0	kW
Maximum Output	13.8	kW
PV Penetration	121	%
Hours of Operation	4,392	hrs/yr
Levelized Cost	0.0723	\$/kWh

PV Power Output

Report Copy Time Series: Plot... Scatter Plot... Delta Plot... Table... Export...

Design and Simulation of a Hybrid Power System for a House in Nigeria:

System Sizing, Optimum Technology and Cost: HOMER Pro:

HOMER Simulation Results:



Simulation Results

System Architecture: Trojan SSIG 12 255 (6.00 strings)
 Jinko Eagle PERC60 300W (14.6 kW) Fronius Galvo 2.5-1 (3.36 kW)
 Autosize Genset (7.20 kW) HOMER Load Following

Total NPC: \$106,307.90
 Levelized COE: \$0.4734
 Operating Cost: \$5,650.04

Cost Summary Cash Flow Compare Economics Electrical Fuel Summary Autosize Genset Renewable Penetration Trojan SSIG 12 255 Jinko Eagle PERC60 300W

Fronius Galvo 2.5-1 Emissions

Quantity	Inverter	Rectifier	Units
Capacity	3.36	3.36	kW
Mean Output	1.70	0.0161	kW
Minimum Output	0	0	kW
Maximum Output	3.36	0.759	kW
Capacity Factor	50.8	0.479	%

Quantity	Inverter	Rectifier	Units
Hours of Operation	8,086	383	hrs/yr
Energy Out	14,935	141	kWh/yr
Energy In	15,721	148	kWh/yr
Losses	786	7.42	kWh/yr

Report Copy Time Series: Plot... Scatter Plot... Delta Plot... Table... Export...

Simulation Results

System Architecture: Trojan SSIG 12 255 (6.00 strings)
 Jinko Eagle PERC60 300W (14.6 kW) Fronius Galvo 2.5-1 (3.36 kW)
 Autosize Genset (7.20 kW) HOMER Load Following

Total NPC: \$106,307.90
 Levelized COE: \$0.4734
 Operating Cost: \$5,650.04

Cost Summary Cash Flow Compare Economics Electrical Fuel Summary Autosize Genset Renewable Penetration Trojan SSIG 12 255 Jinko Eagle PERC60 300W

Fronius Galvo 2.5-1 Emissions

Quantity	Value	Units
Batteries	24.0	qty.
String Size	4.00	batteries
Strings in Parallel	6.00	strings
Bus Voltage	48.0	V

Quantity	Value	Units
Autonomy	29.9	hr
Storage Wear Cost	0.302	\$/kWh
Nominal Capacity	74.1	kWh
Usable Nominal Capacity	59.3	kWh
Lifetime Throughput	44,448	kWh
Expected Life	5.02	yr

Quantity	Value	Units
Average Energy Cost	0	\$/kWh
Energy In	9,875	kWh/yr
Energy Out	7,927	kWh/yr
Storage Depletion	30.0	kWh/yr
Losses	1,978	kWh/yr
Annual Throughput	8,863	kWh/yr

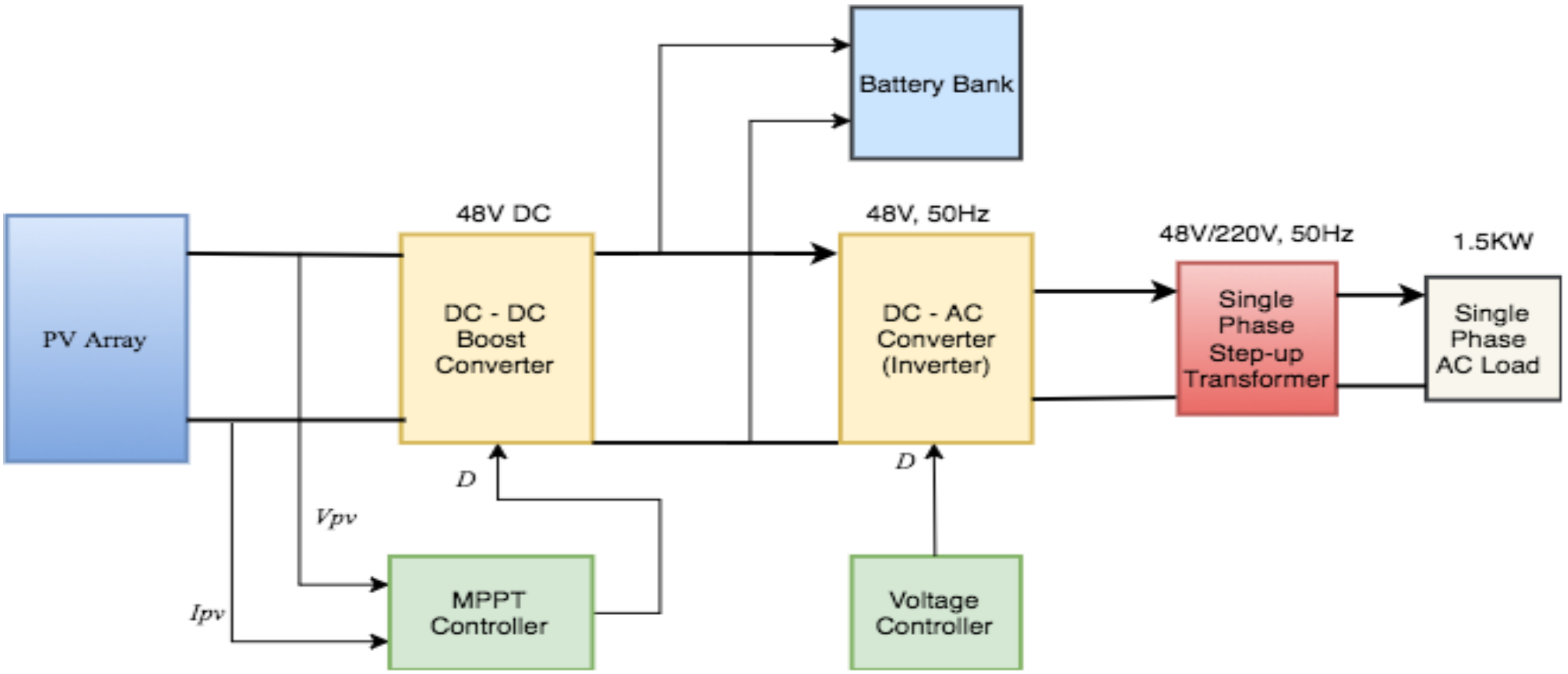
Report Copy Time Series: Plot... Scatter Plot... Delta Plot... Table... Export...

Design and Simulation of a Hybrid Power System for a House in Nigeria:



Dynamic/Mathematical model of each of the subsystems (components):

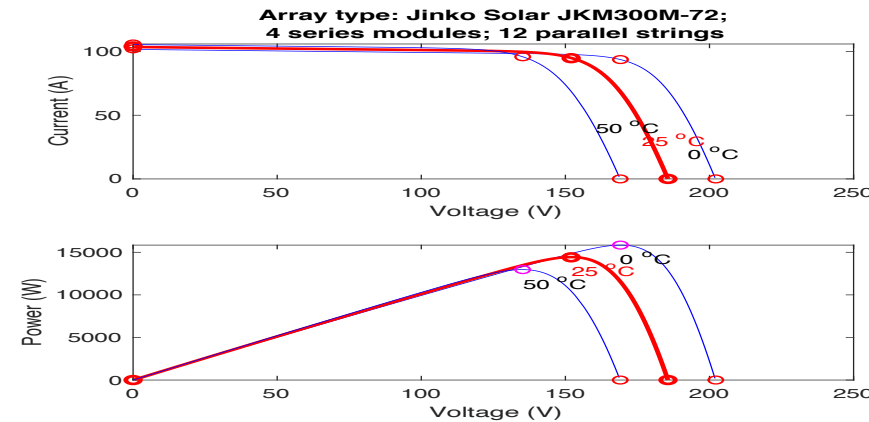
Block Diagram of PV System Components:



Design and Simulation of a Hybrid Power System for a House in Nigeria:

Dynamic/Mathematical model of each of the subsystems (components): Overview

- Jinko Solar JMK300M-72 PV Array: 4 strings of 12 panels PV modules. ($12 \times 4 \times 300 = 14.4 \text{KW}$)



- The converter stabilizes and steps up (boosts) the DC Voltage from the PV Array to a fixed DC Voltage output.
- The converter provides the DC bus voltage (48V) for charging the Battery.
- The output voltage of the DC – DC Boost Converter is fed into the Inverter for conversion to AC voltage.
- The single phase voltage source inverter converts the fixed DC Voltage (48V) into a single-phase AC voltage (48V) with fixed frequency of 50Hz.
- The single-phase Step-up transformer steps up the 48V Inverter output to 220V AC at 50Hz for household loads
- Trojan SSIG 12 255 Lead Acid Batteries: 48V Bus Voltage, 1542Ah Capacity, 29.9 hrs Autonomy,

Design and Simulation of a Hybrid Power System for a House in Nigeria:

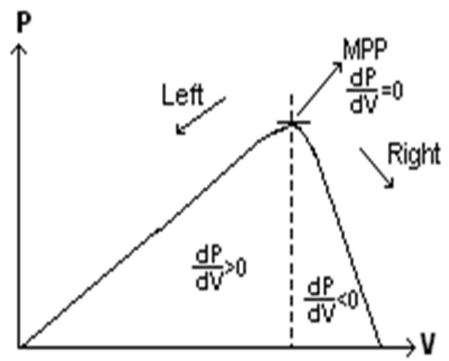


Dynamic/Mathematical model of each of the subsystems (components): Controls

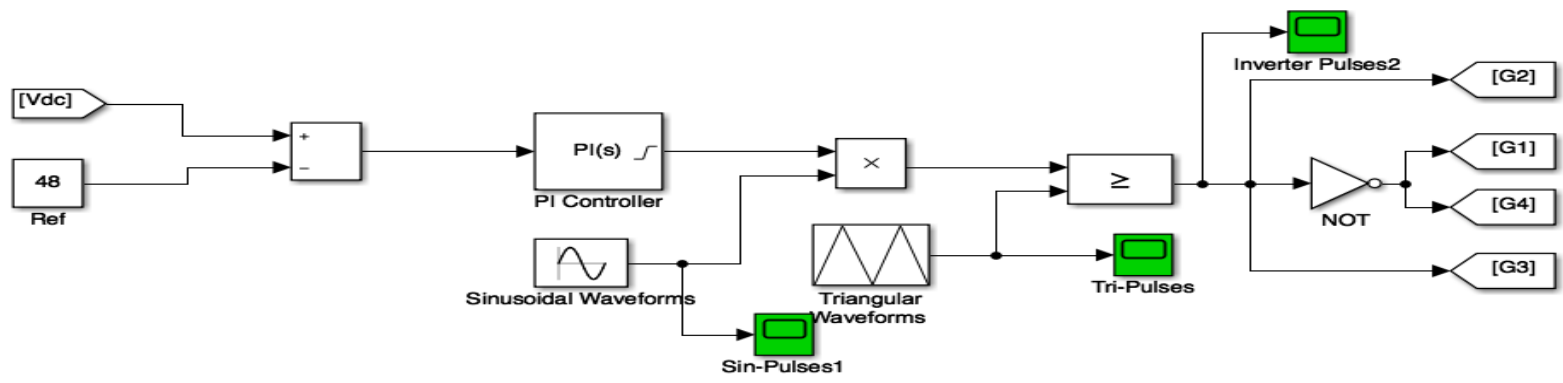
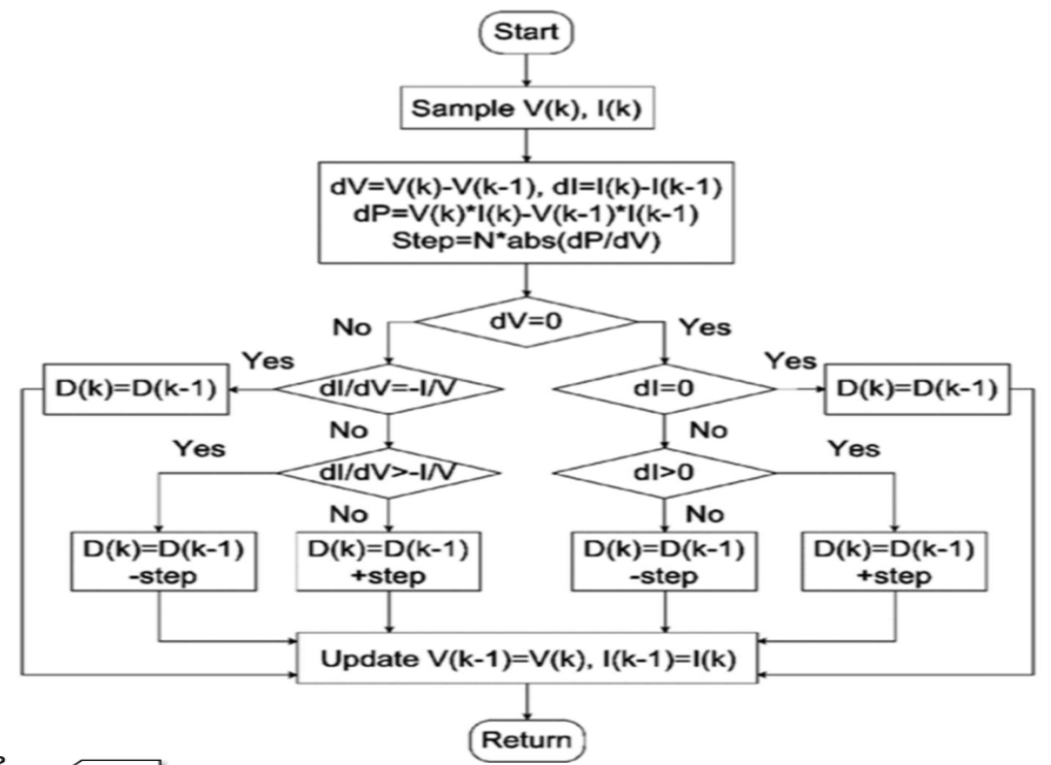
Incremental Conductance MPPT Algorithm and Inverter Voltage Control:

$$P = V * I$$

$$\frac{dP}{dV} = \frac{d(I * V)}{dV} = V * \frac{dI}{dV} + I$$

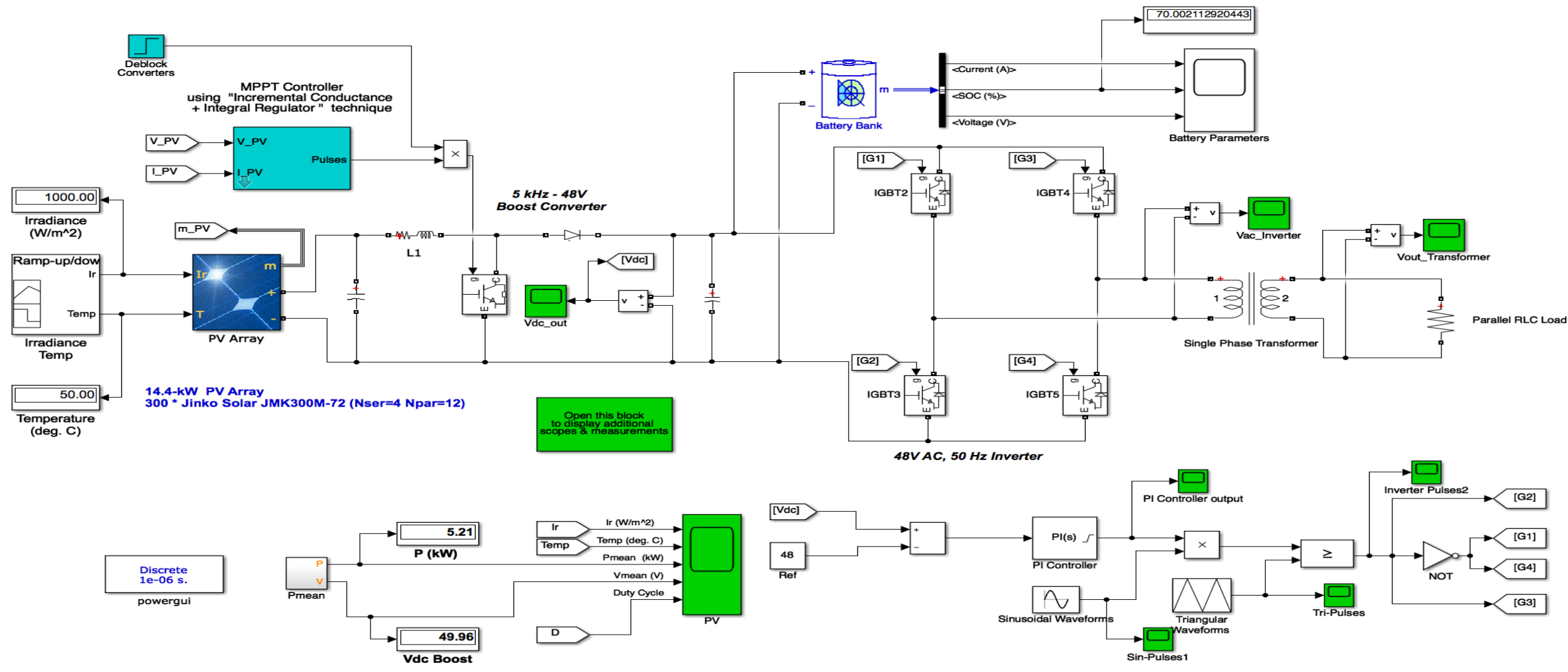


- dI/dV = incremental conductance. & I/V = panel conductance
- Incremental and Panel conductance are determined.
- DC – DC Boost Converter Duty Cycle, D is adjusted until:
- $dI/dV = -I/V$ or $dP/dV=0 =$ MPPT



Design and Simulation of a Hybrid Power System for a House in Nigeria:

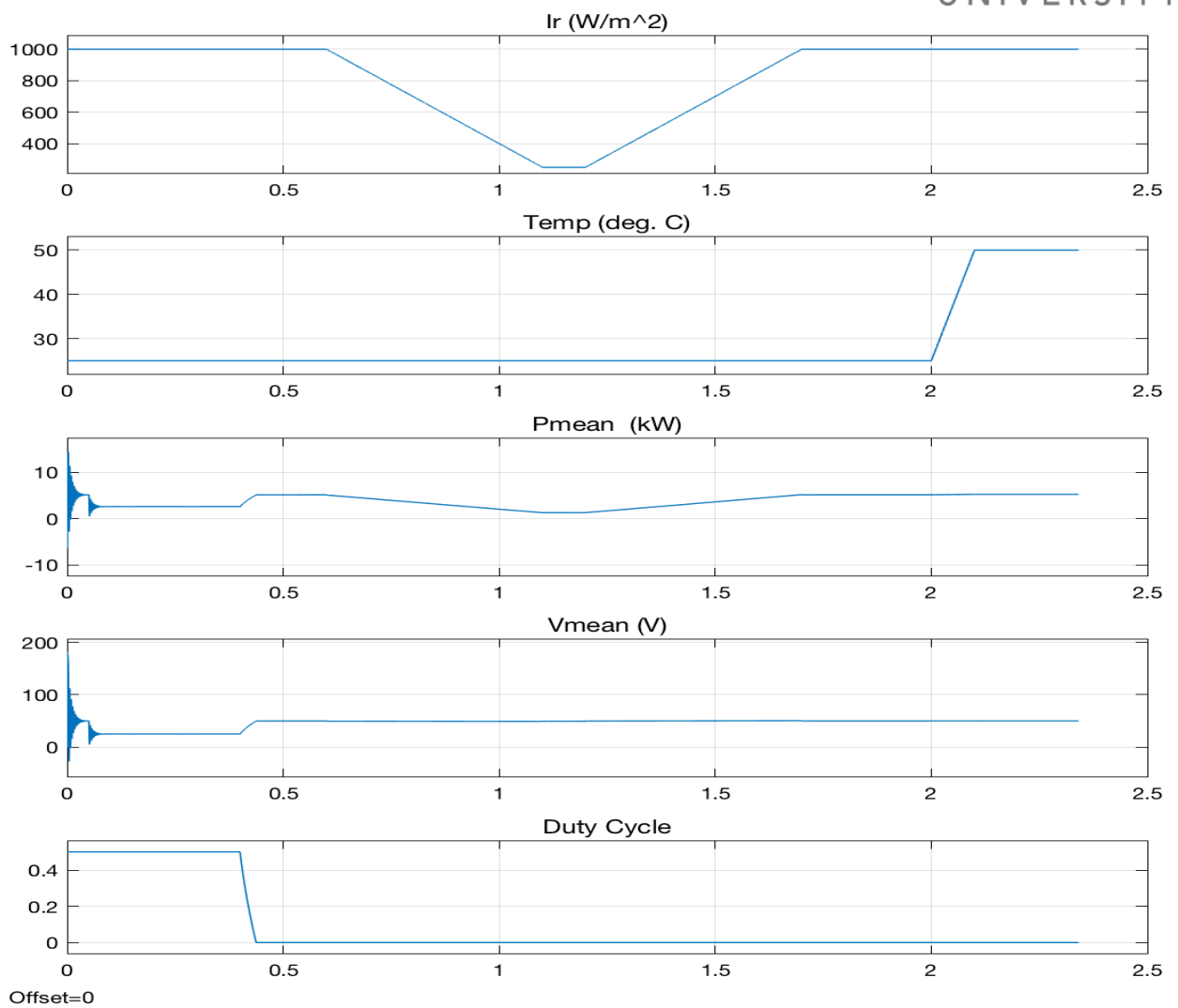
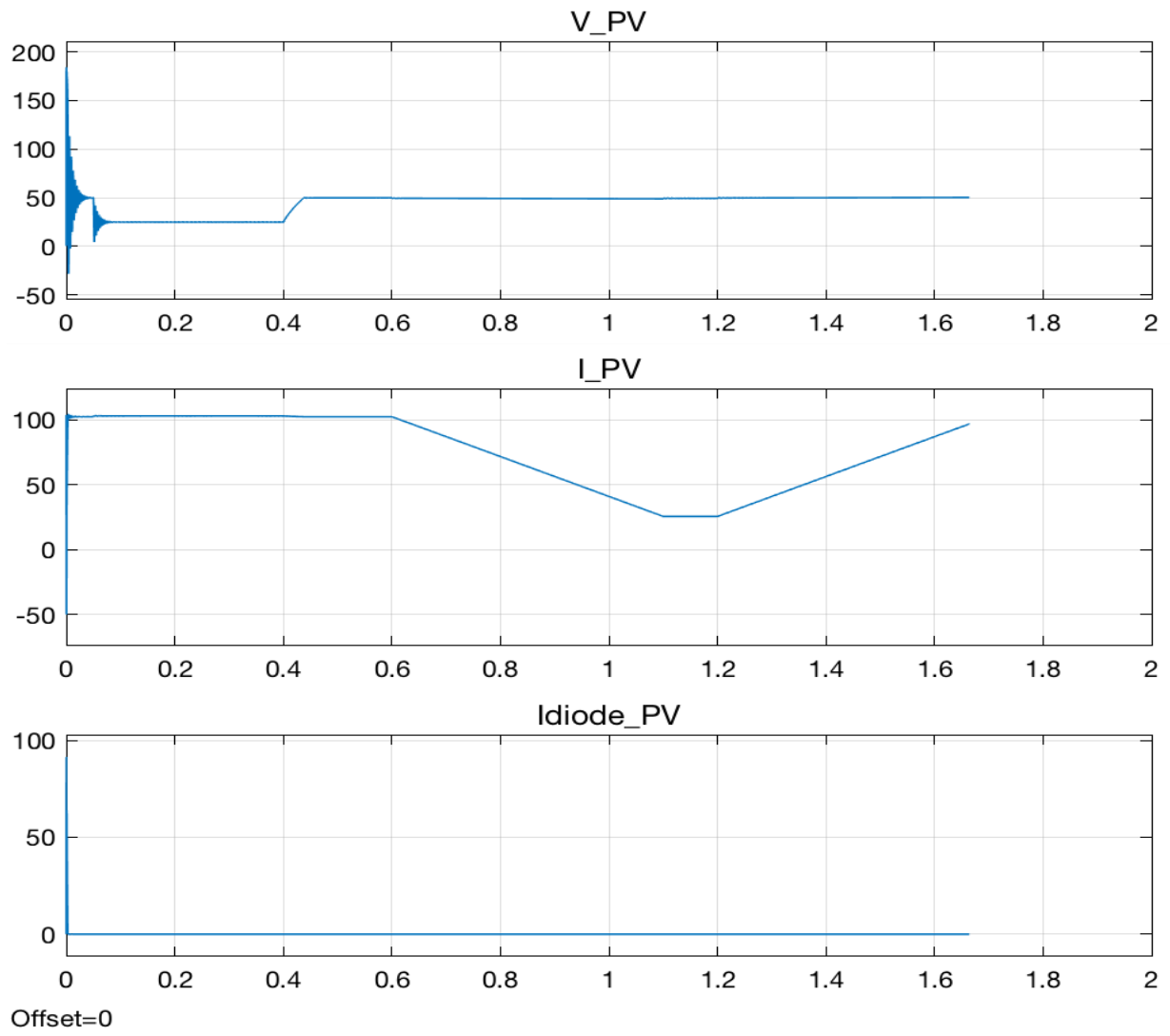
Matlab/Simulink Simulation of the PV System Components of the HPS



Design and Simulation of a Hybrid Power System for a House in Nigeria:



Simulation Results:



Design and Simulation of a Hybrid Power System for a House in Nigeria:



Simulation Results:

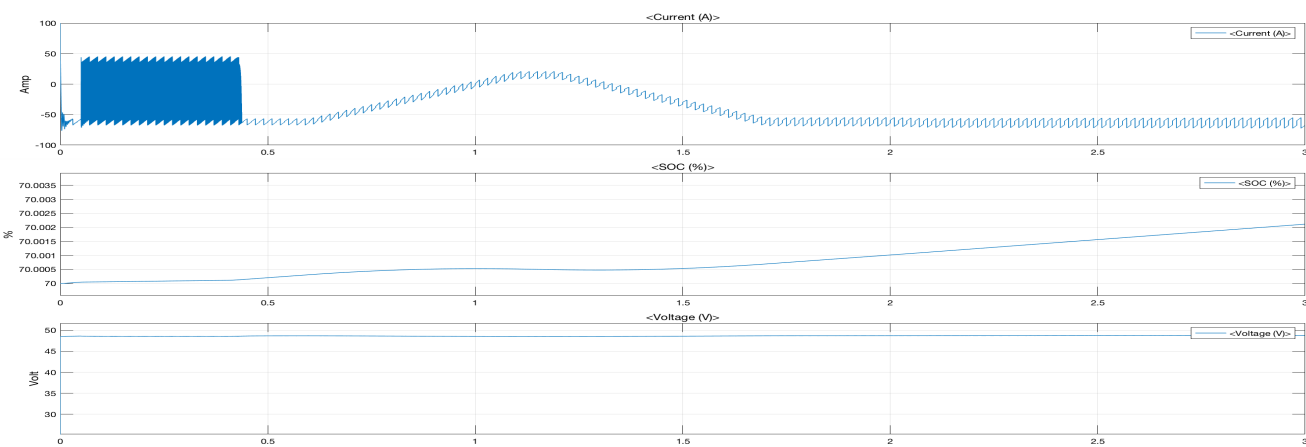


Fig.: Battery I, SOC & V

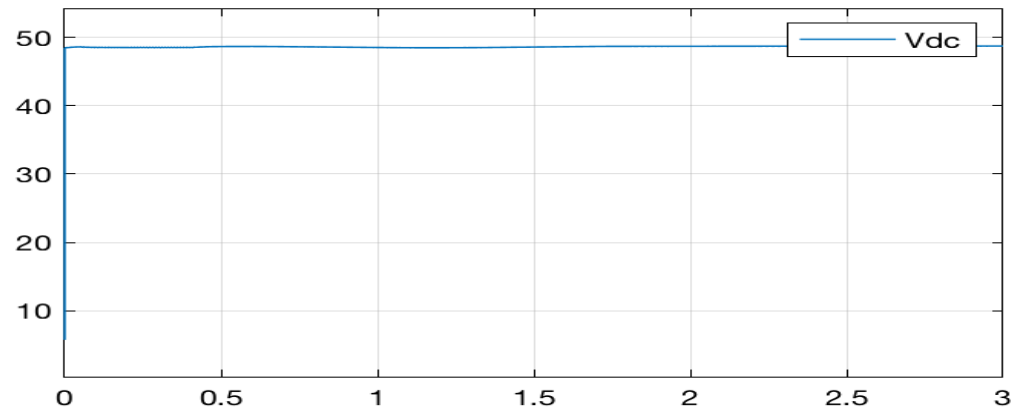


Fig.: DC – DC Boost Converter Output Voltage

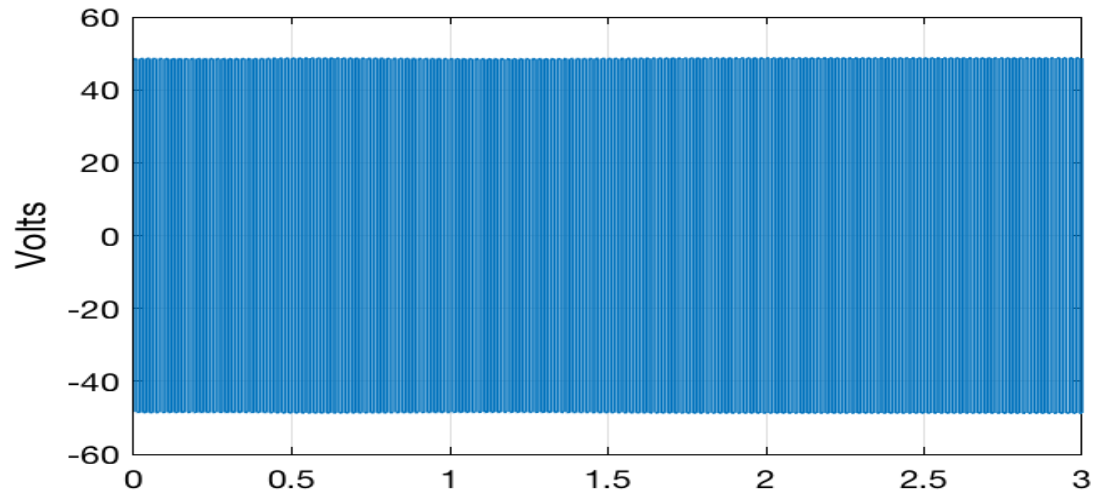


Fig.: Inverter Output Voltage

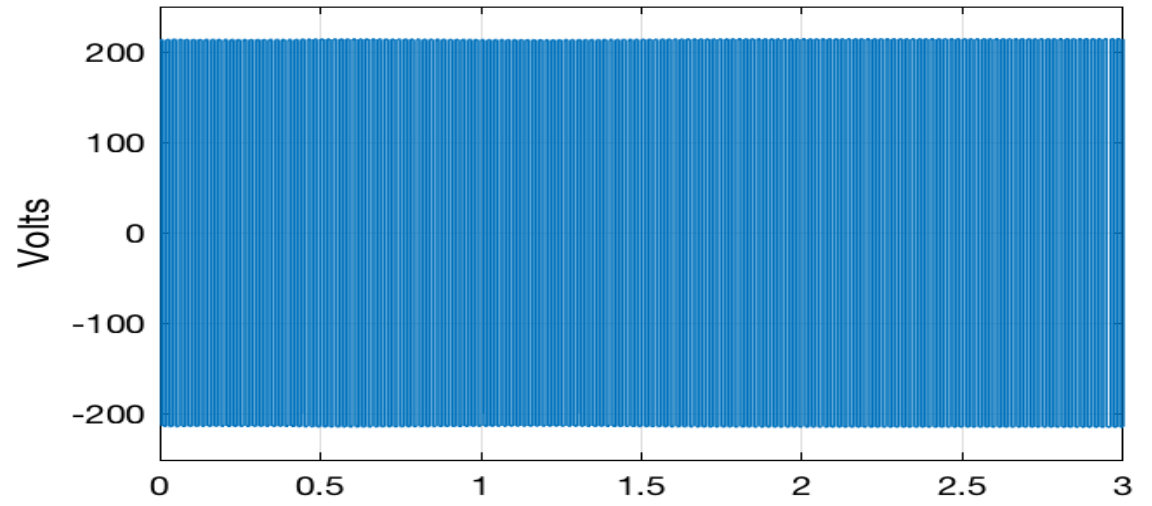
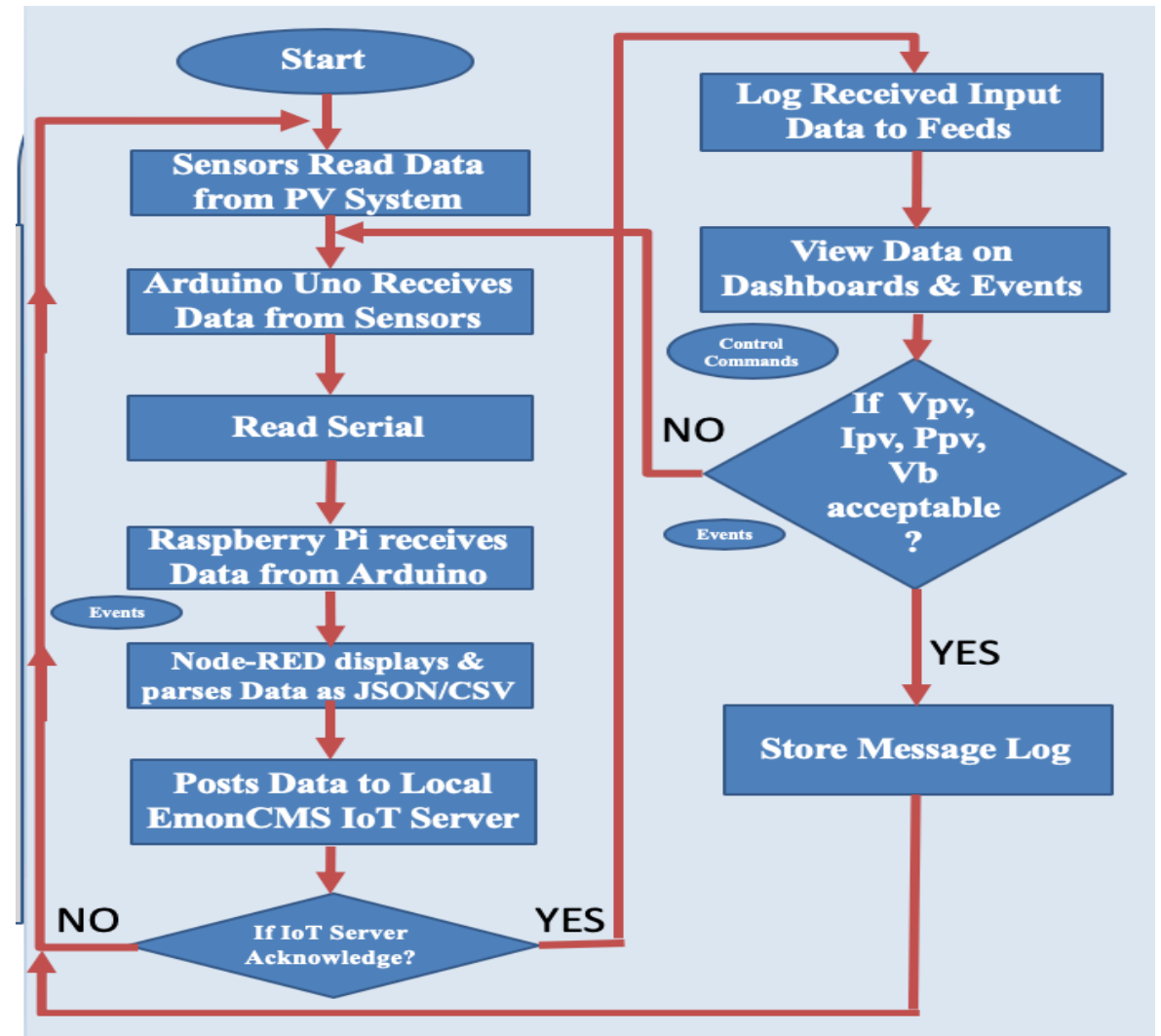
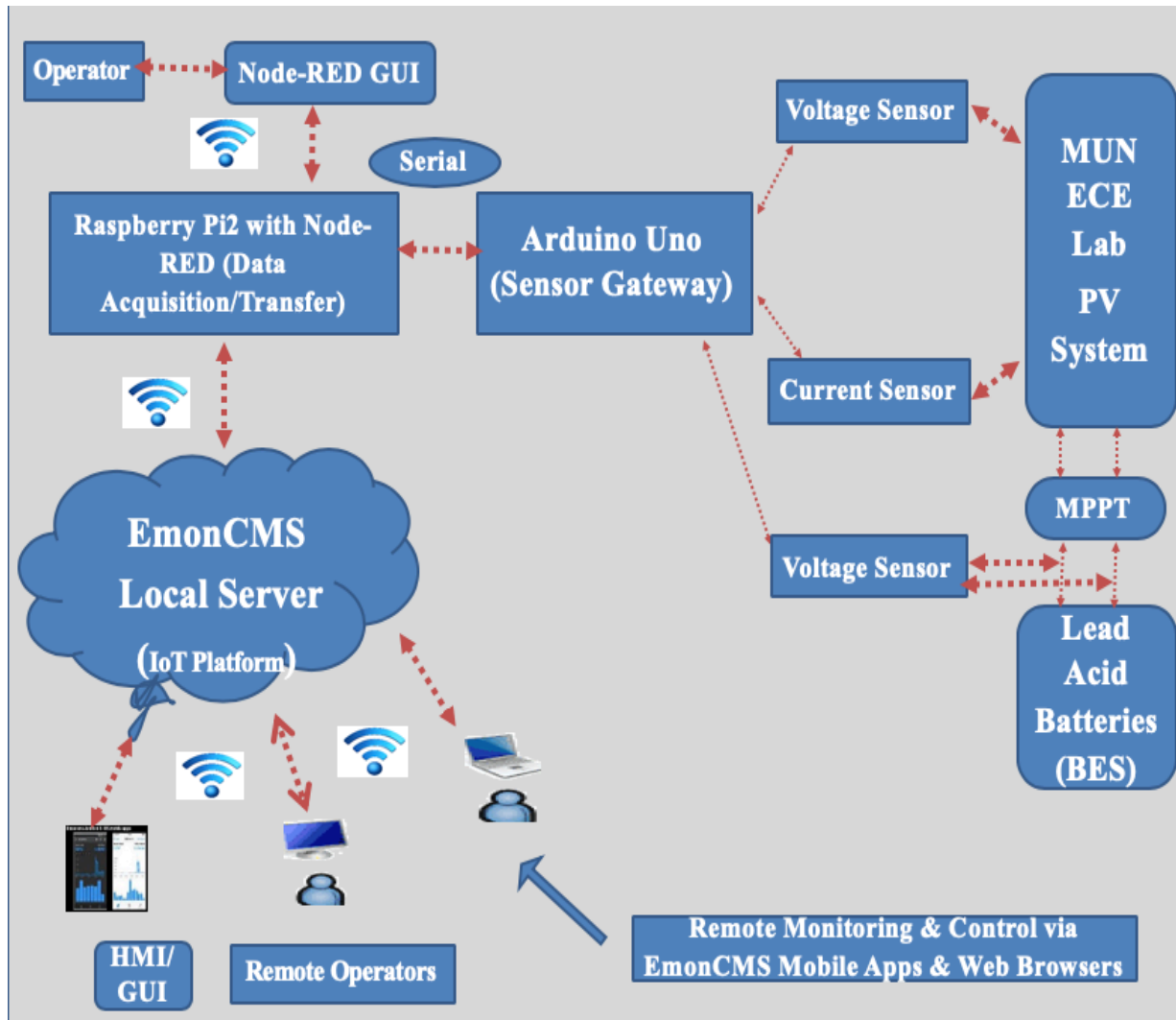


Fig.: Transformer Output Voltage: Load Voltage

Components of SCADA 1 System:

- A Small Renewable Power Generation System (Solar PV System with ESS).
- Sensors (Current, Voltage.): FIDs
- Arduino Uno R3 Microcontroller: RTU
- Raspberry Pi2 Single-Board Computer: RTU
- Node-RED Programming Tool: SCADA Communication/Data Transfer
- EMONCMS Local Server IoT Platform: MTU
- Computers and Mobile Devices: HMIs/GUIs

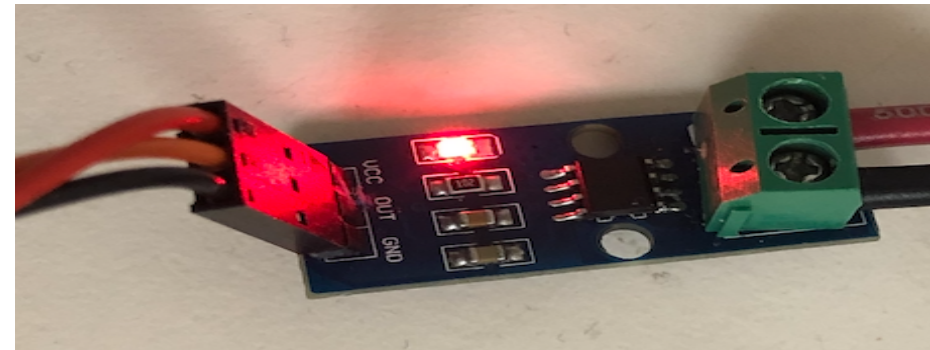
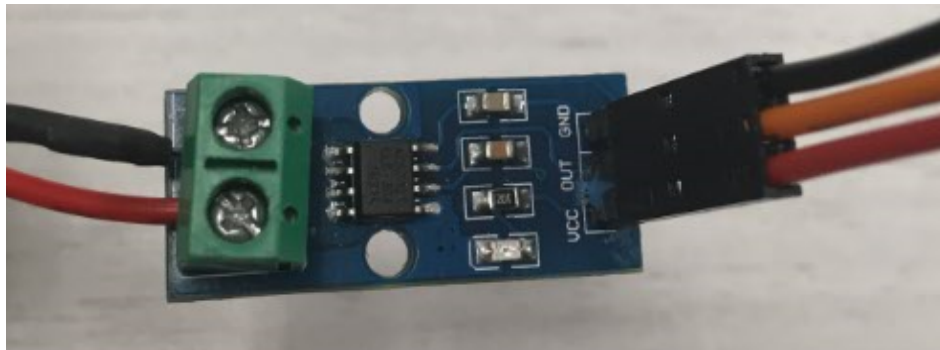
System Configuration & Flow Chart:



Field Instrumentation Devices: Sensors

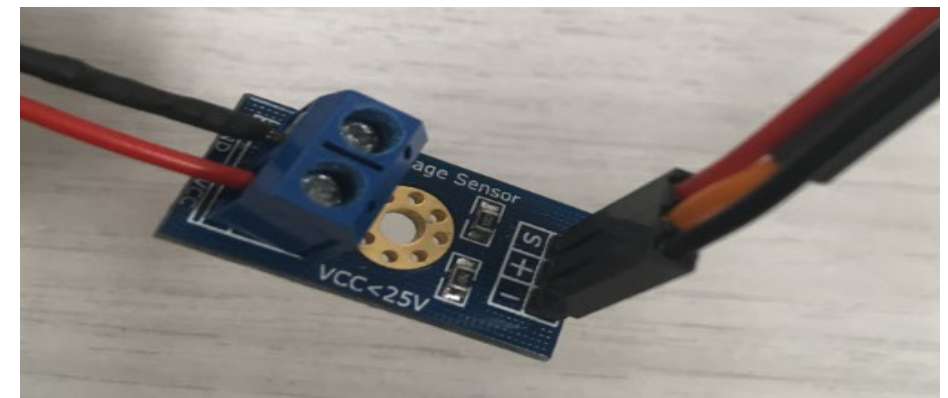
The ACS 712 Hall Effect Current Sensor:

- Hall Effect-Based Linear Current Sensor, 5.0 V single supply operation, 66 to 185 mV/A output sensitivity, Low cost
- Output voltage proportional to AC or DC currents, Nearly zero magnetic hysteresis
- Used to measure the DC Current from the solar PV System (the 30A Model was used for this application)



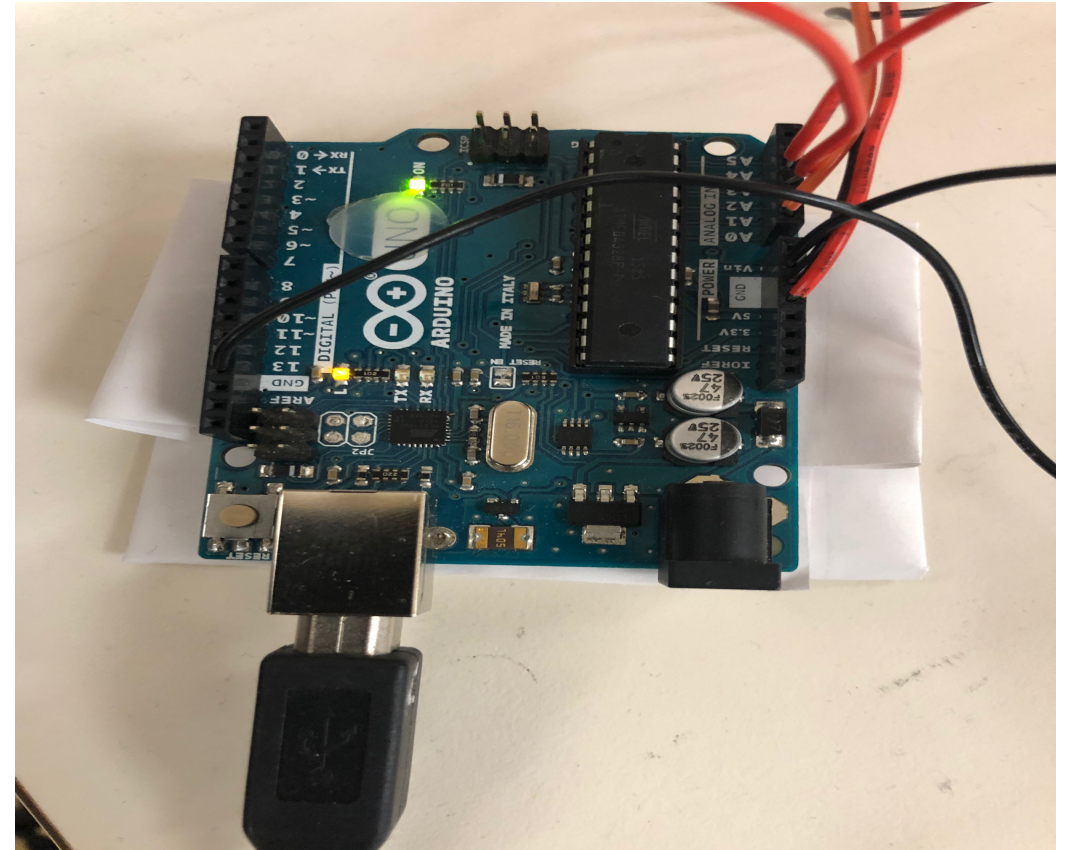
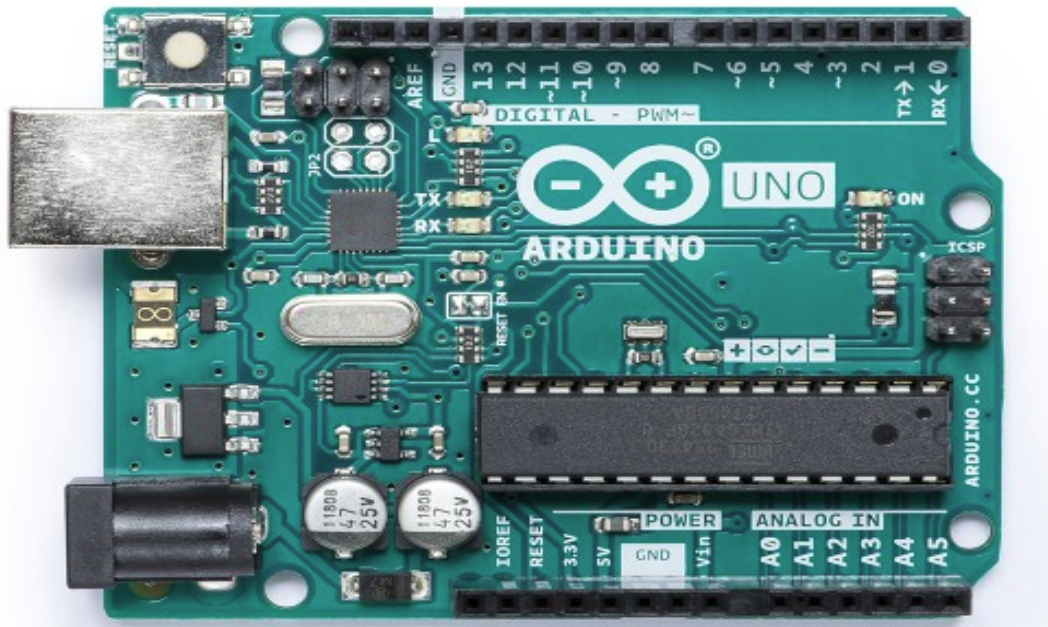
The MH Electronic Voltage Sensor Module:

- Two modules used.
- Detects the supply voltage from 0.025 to 25V
- Analog input, Low Cost
- Uses the concept of Voltage Divider to measure the supply voltage



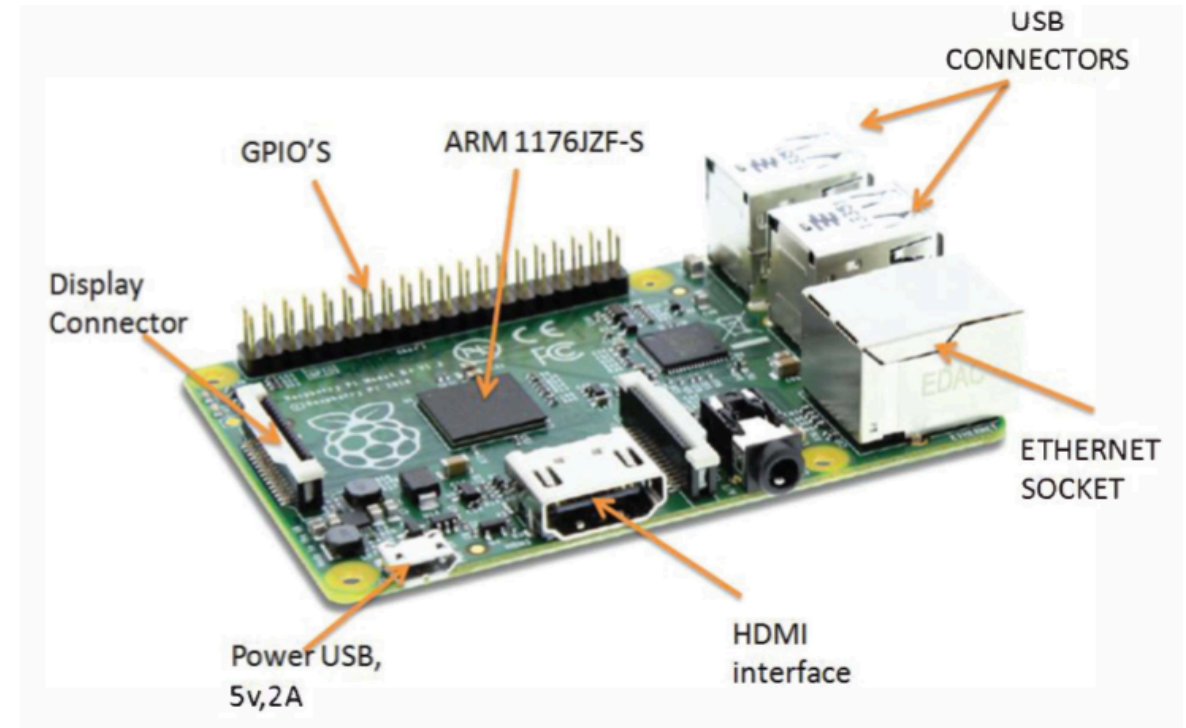
Arduino Uno R3 Microcontroller: RTU

- A microcontroller board based on the ATmega328P; Version R3 used
- 14 digital input/output pins, 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button.
- Programmed to measure DC Current & Voltage from the PV Panel.



Raspberry Pi Single-Board Computer: RTU

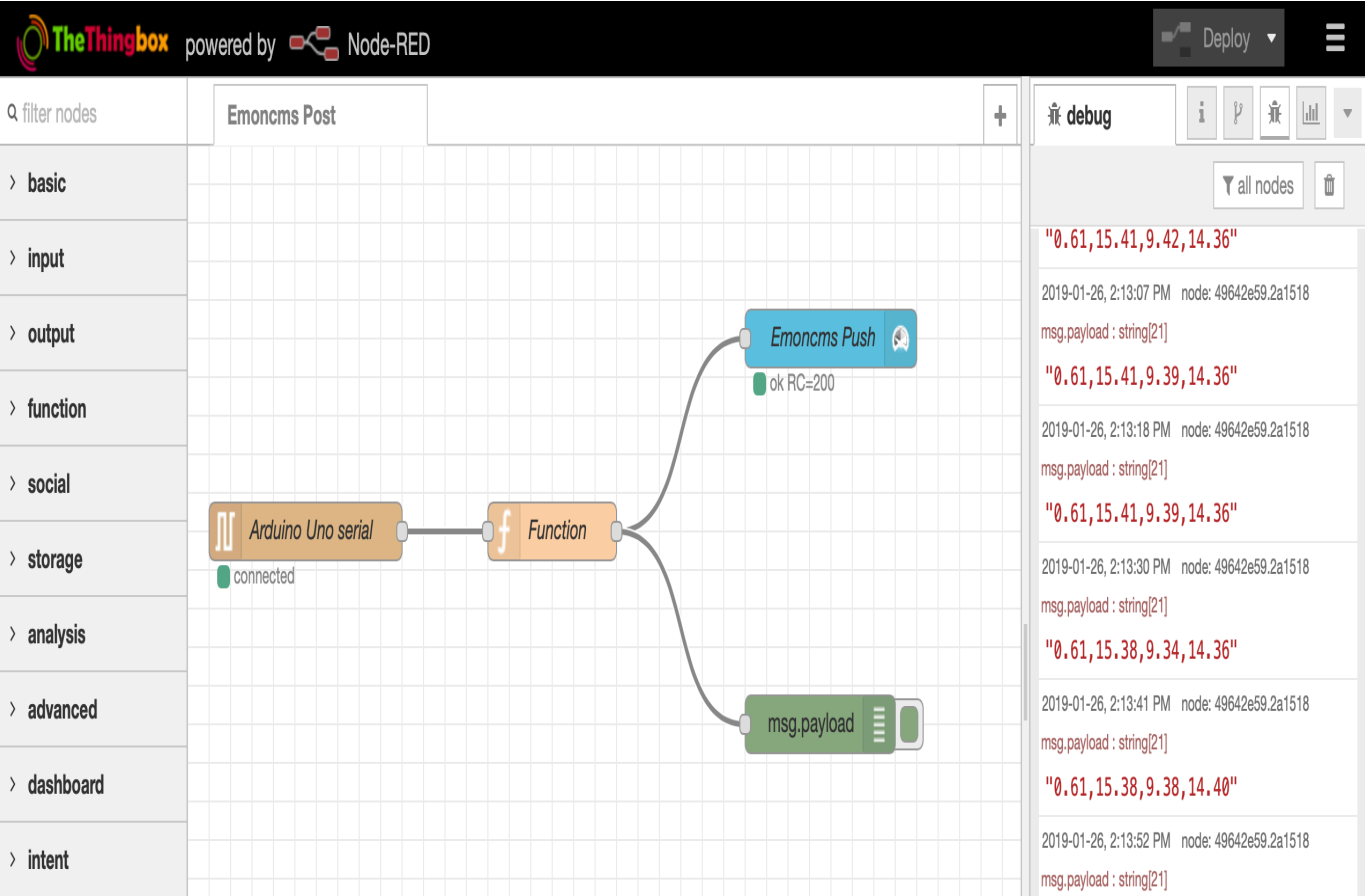
- A single-board computer device with BCM2836 quad core ARMv7 processor
- Hosted the Node-RED Programming Tool for parsing data to Emoncms IoT Platform
- Connected to MUN Network via an Ethernet cable.
- Arduino Uno board connected to the RPi via a USB cable.



SCADA Communication Channel: Ethernet + Node-RED

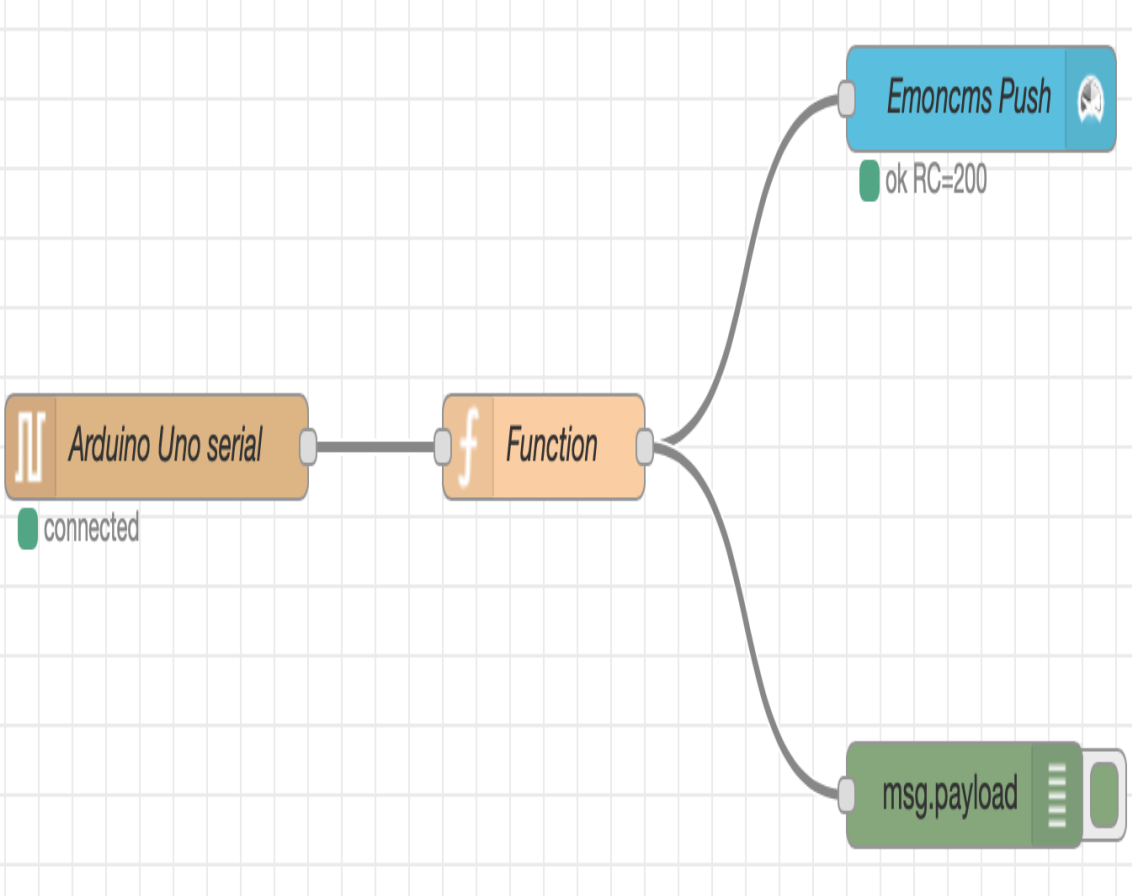
Node-RED Programming Tool:

- An open source programming tool for wiring together hardware devices, APIs & online services (IoT)
- Flows written to acquire Arduino Sensor Data via Serial, and Post to Emoncms IoT Server.



The screenshot shows the Node-RED web interface. On the left, a sidebar lists various node categories. The main workspace contains a flow with three nodes: 'Arduino Uno serial' (connected), 'Function', and 'Emoncms Push' (ok RC=200). A 'debug' console on the right displays a series of JSON messages from the 'Function' node, each containing an array of sensor data values.

```
debug console output:  
"0.61,15.41,9.42,14.36"  
2019-01-26, 2:13:07 PM node: 49642e59.2a1518  
msg.payload : string[21]  
"0.61,15.41,9.39,14.36"  
2019-01-26, 2:13:18 PM node: 49642e59.2a1518  
msg.payload : string[21]  
"0.61,15.41,9.39,14.36"  
2019-01-26, 2:13:30 PM node: 49642e59.2a1518  
msg.payload : string[21]  
"0.61,15.38,9.34,14.36"  
2019-01-26, 2:13:41 PM node: 49642e59.2a1518  
msg.payload : string[21]  
"0.61,15.38,9.38,14.40"  
2019-01-26, 2:13:52 PM node: 49642e59.2a1518  
msg.payload : string[21]
```

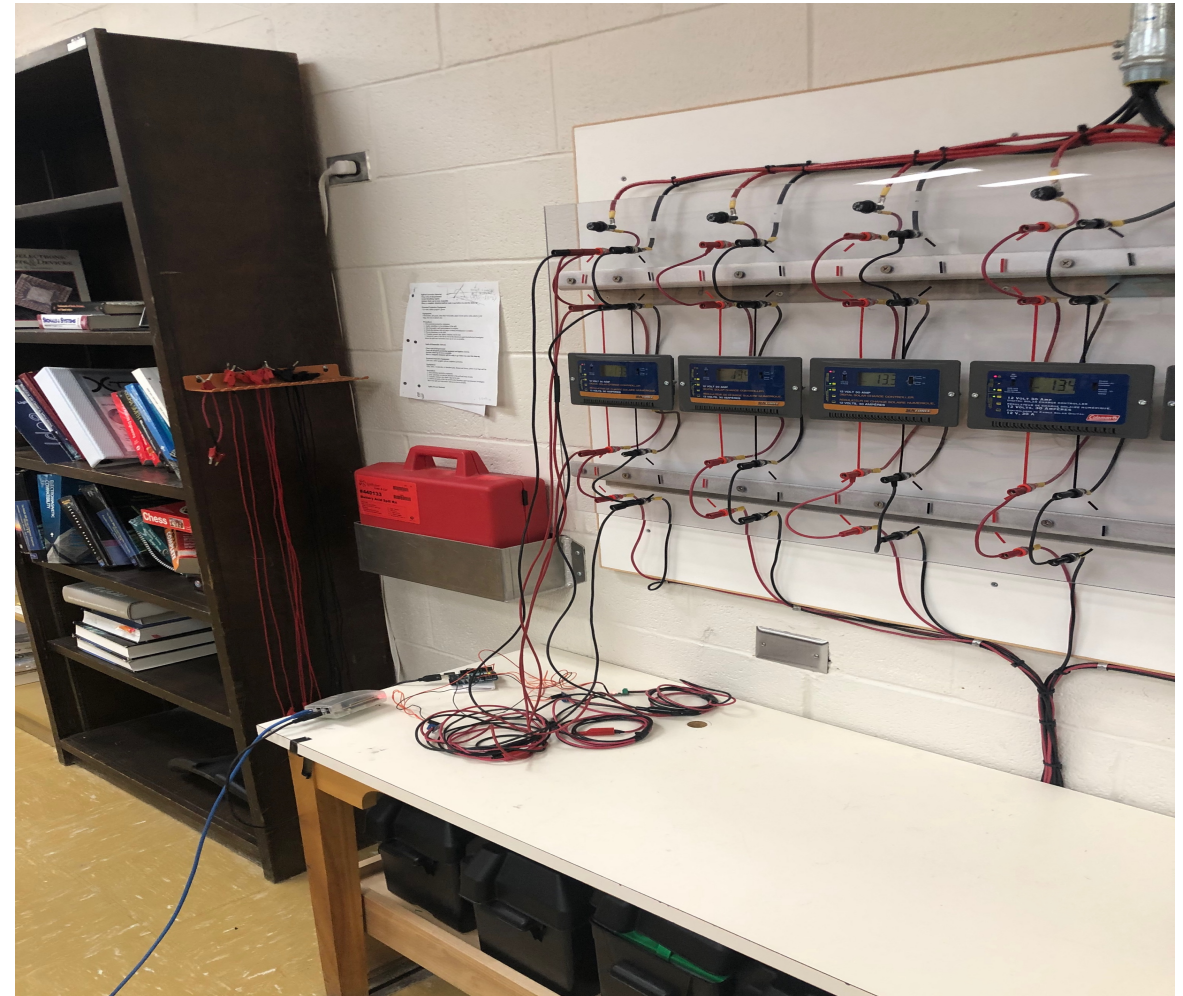
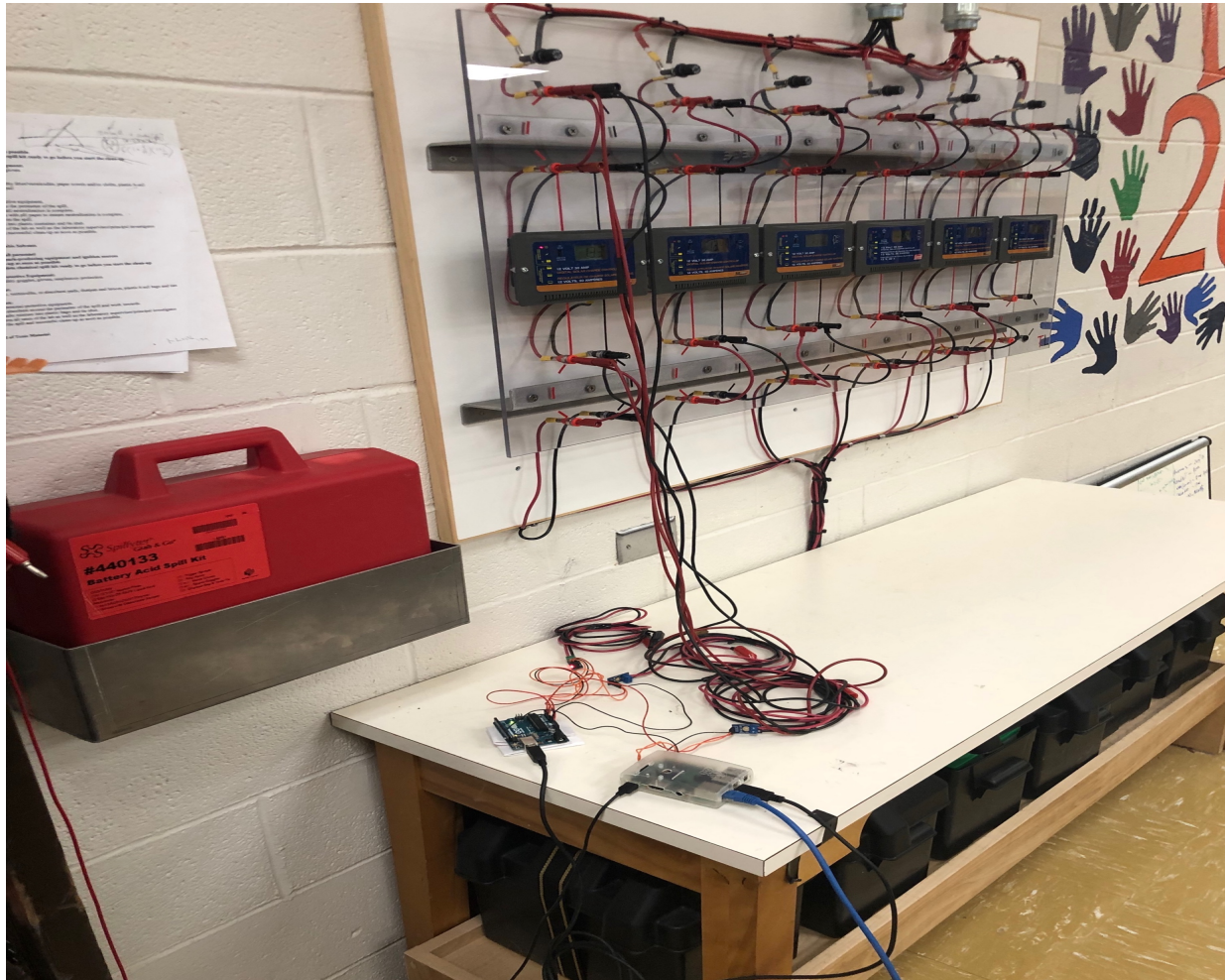


EMONCMS Local Server IoT Platform (MTU): Introduction to EMONCMS

- A powerful open-source web-app for processing, logging & visualising energy data, temperature, etc.
- Allows Users to create customized reports in the form of charts, data logs & alarms (local, email, web, and mobile apps) for remote data monitoring and control.
- Part of the OpenEnergyMonitor.org project.
- Both Local and Web-Server options available.
- Free Local Server Software option for various Operating Systems.
- Locally installed on Jetson TK1 Dev. Kit Linux Machine; Data Security technique.
- Server Self-managed and Self-hosted on Private Network (MUN Network); Data Security technique.

Hardware Implementation & Experimental Setup on a Small Renewable Power Generation System:

- Hardware Prototype designed & set up in MUN ECE Laboratory for PV System Monitoring & Supervisory Control:



Acquired PV System Data on Emoncms IoT Platform:





















Dashboards Apps Extra Setup Logout

Inputs Input API Help

+ Node 1 inactive					
- Node 10 5s					
Node	Key	Name	Process list	Updated	Value
10	1	PV Current	Log to feed	5s	0.6   
10	2	PV Voltage	Log to feed	5s	15.48   
10	3	PV Power	Log to feed	5s	9.28   
10	4	Battery Voltage	Log to feed	5s	14.4   

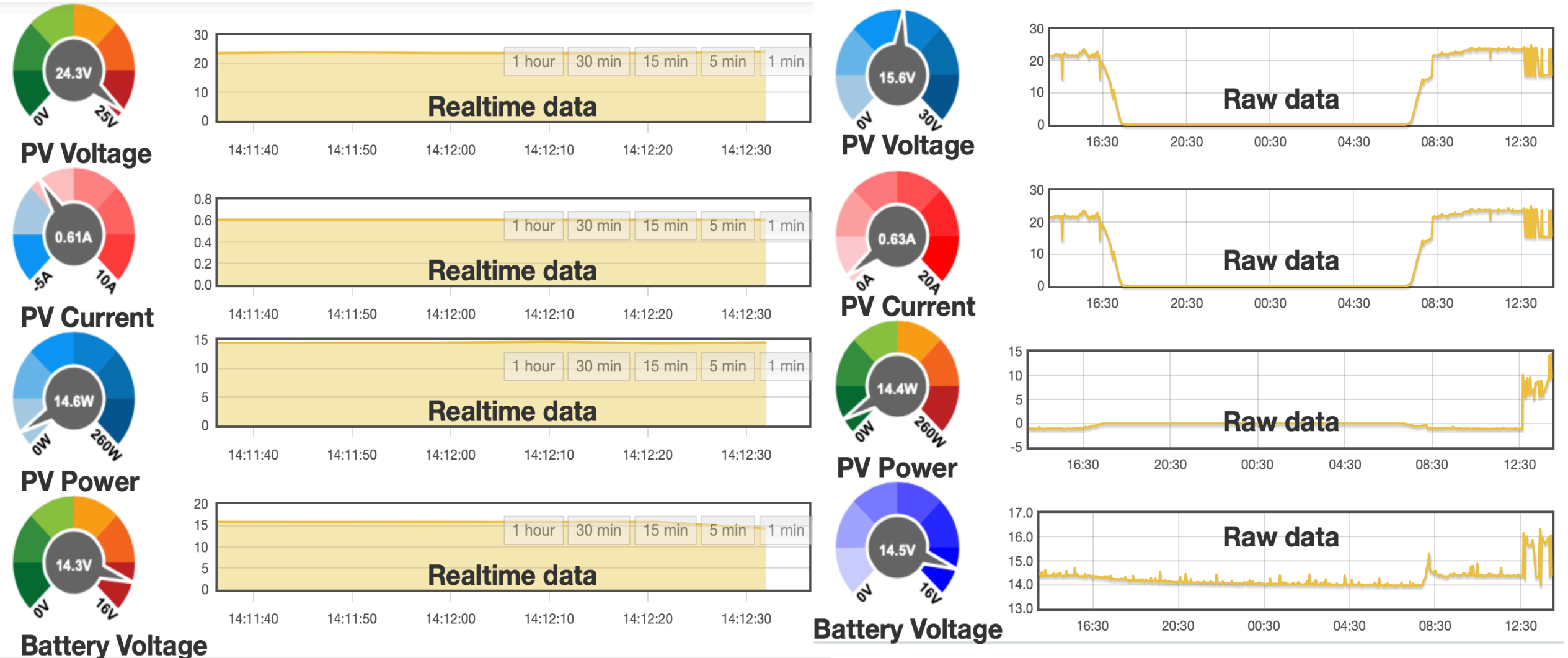
Dashboards Apps Extra Setup Logout

Feeds Feed API Help

+ Node 1 301KB inactive									
- Node 10 0B 8s									
Id	Tag	Name	Process list	Public	Datatype	Engine	Size	Updated	Value
24	...	PV Power_W			REALTIME	PHPFINA	n/a	8s	9.28    
15	...	PV Current_A			REALTIME	PHPFINA	n/a	8s	0.6    
19	...	PV Voltage_V			REALTIME	PHPFINA	n/a	8s	15.5    
25	...	Battery Voltage_V			REALTIME	PHPFINA	n/a	8s	14.4    

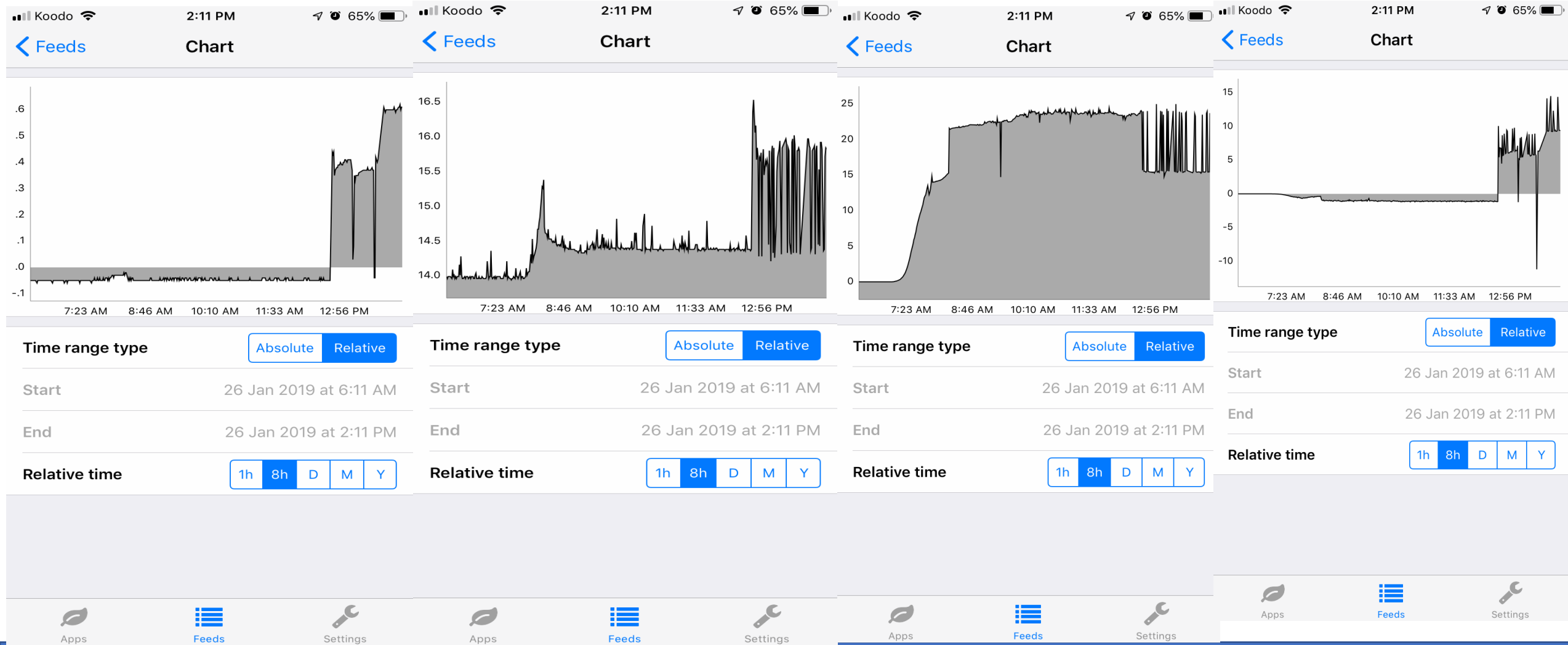
[Refresh feed size](#) [New virtual feed](#)

Human Machine Interface: Operator Dashboards and Events: Emoncms IoT Platform:



Human Machine Interface: Operator Dashboards and Events: Emoncms IoT Mobile App Platform:

- Screenshots of PV Current, Voltage & Power from EMONCMS Mobile App:



The image displays four screenshots of the Emoncms mobile app interface, each showing a different data chart for PV Current, Voltage, and Power. Each screenshot includes a status bar at the top with 'Koodo' carrier, signal strength, Wi-Fi, and 65% battery. The app title 'Feeds' and 'Chart' are visible. The charts show data from 7:23 AM to 12:56 PM on 26 Jan 2019. Below each chart are controls for 'Time range type' (Absolute/Relative), 'Start' and 'End' times, and 'Relative time' (1h, 8h, D, M, Y). A bottom navigation bar contains 'Apps', 'Feeds', and 'Settings' icons.

Conclusions: Some Key Features of SCADA 1:

- IoT-based, Secure, Reliable, Low-cost, Low-Power, & Open Source
- Reporting, Remote Data Monitoring & Supervisory Control, Data Acquisition, Historic Data Storage & Trending, etc.

Table A.1: Bill of Materials.

S/N	COMPONENT	QTY	PRICE (CAD)
1	Emoncms Server (Jetson TK1 Dev. Kit)	1	250.00
2	Emoncms Software	1	00.00
3	Node-RED Software	1	00.00
4	Raspberry Pi 2 B	1	45.95
5	Arduino Uno	1	29.00
6	Current Sensor	1	5.25
7	Voltage Sensor	2	11.98
8	8GB SD Card	1	12.66
9	Miscellaneous (Wires, Boxes, etc.)	1	50.00
Grand Total:			\$ 404.84 CAD

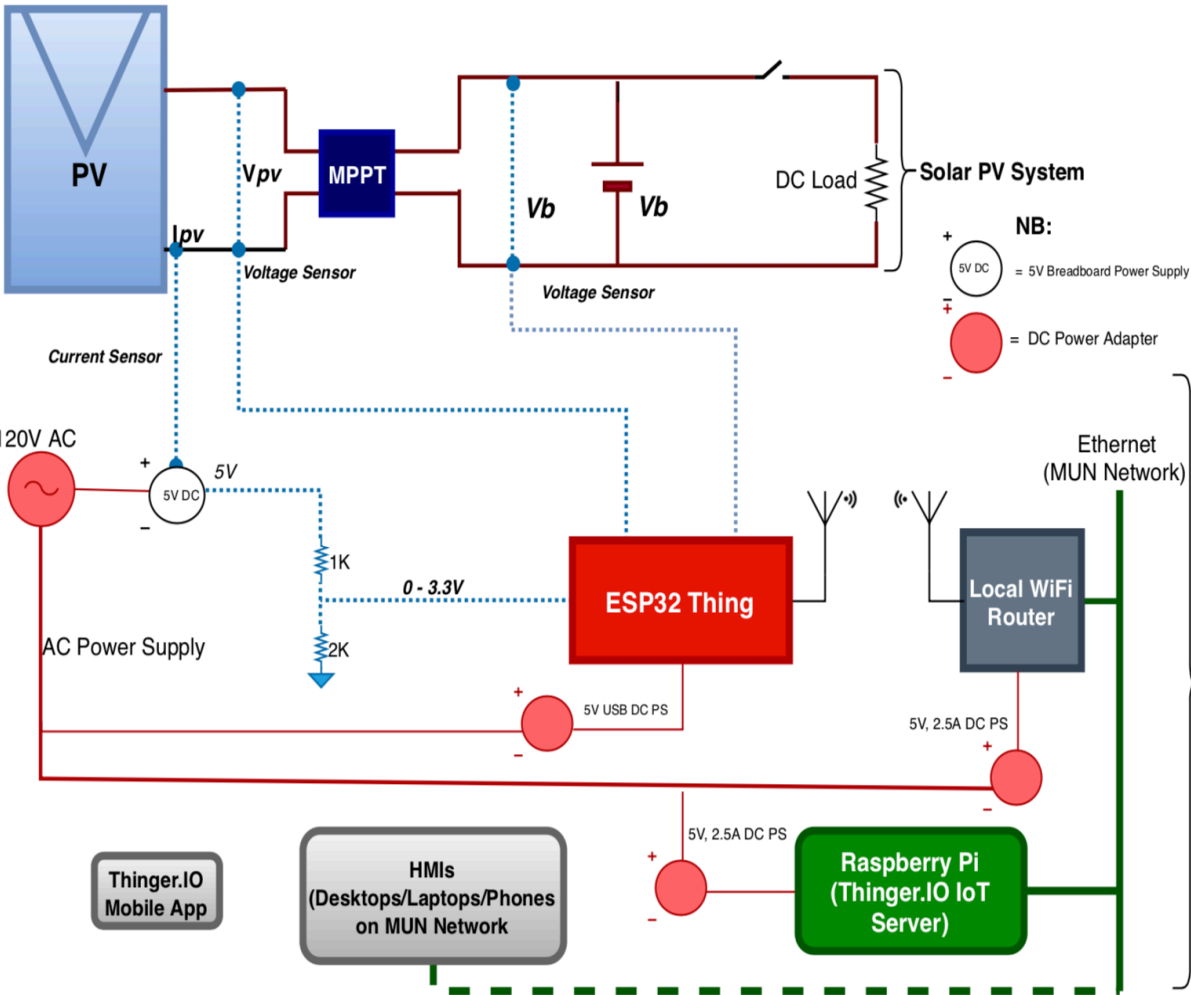
Table A.2: Power consumption of hardware components.

S/N	HARDWARE	POWER (W)
1	Raspberry Pi 2 + Arduino Uno + Sensors	2.4
2	Emoncms Server	2.7
Total Power Consumption :		5.1 W

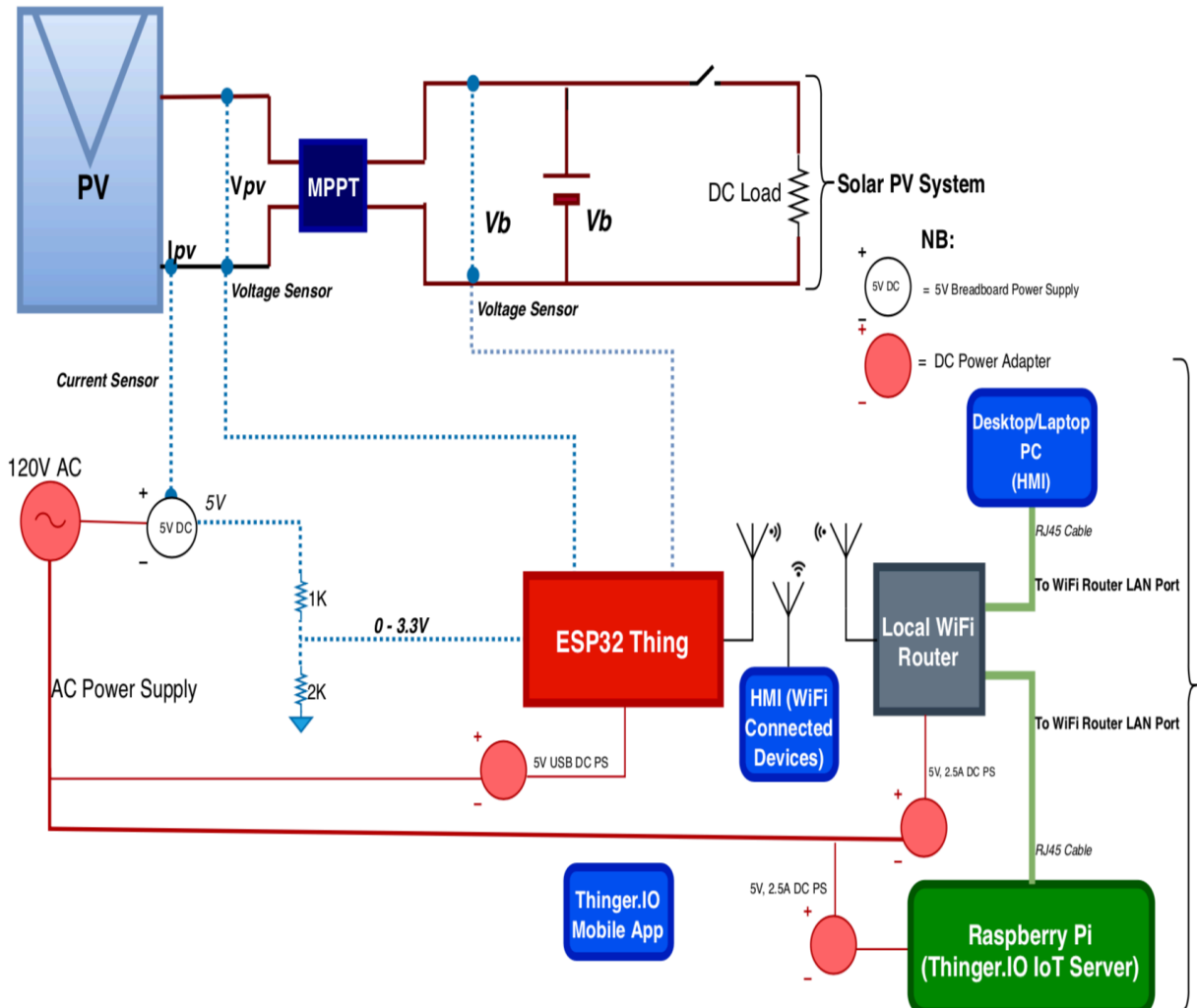
Components of SCADA 2 System:

- A Small Renewable Power Generation System (Solar PV System with ESS).
- Sensors (Current, Voltage.): FIDs
- ESP32 Thing Micro-controller: RTU
- Raspberry Pi2 machine (Thinger.IO IoT Server): MTU
- Wi-Fi Router: SCADA Communication/Data Transfer
- Computers and Mobile Devices: HMIs/GUIs

System Configurations; Two Configurations considered



Configuration A: MUN Network (Internet Required)

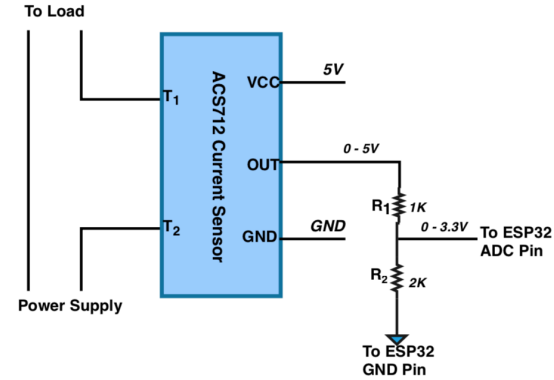


Configuration B: Private Industrial Network created (Internet Optional)

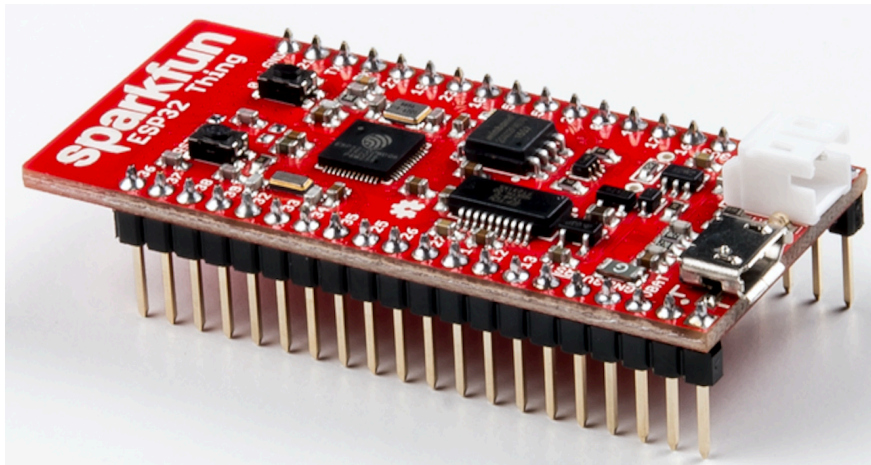
Field Instrumentation Devices: Same Sensors as in SCADA 1

ESP32 Thing Micro-controller: RTU

- Low-Cost (\$20), Low-Power (0.5W) Micro-controller for IoT Projects
- Supports Wi-Fi, 30 Input/Output Pins
- 5 V Power Supply, 2.2 V to 3.6 V ((3.3 V) ADC Signal Voltage
- Programmed with Arduino IDE to measure & acquire Sensor PV Data & parse data over Wi-Fi to Thinger.IO IoT Server



SparkFun ESP32 Thing (DEV-13907)



Name	ADC
Power	DAC
GND	SPI
Control	UART
Arduino	Touch
GPIO Port	Misc

*GPIO: Port Input Only
*ADC: Preamplifier ADC
GPIO 3.3V tolerant only

Reset button	Button: GPIO 0
ADC1_0*	GPIO36*
ADC1_1*	GPIO37*
ADC1_2*	GPIO38*
ADC1_3*	GPIO39*
ADC1_4	GPIO32
ADC1_5	GPIO33
VDET1	ADC1_6
VDET2	ADC1_7
DAC1	ADC2_8
DAC2	ADC2_9
Touch9	ADC1_4
Touch8	ADC1_5
Touch7	ADC2_7
Touch6	HSPI_CLK
Touch5	HSPI_Q
Touch4	HSPI_ID
GPIO36*	SenseVP
GPIO37*	CapVP
GPIO38*	CapVN
GPIO39*	SensVN
GPIO32	XTAL32
GPIO33	XTAL32
GPIO34*	
GPIO35*	
GPIO25	
GPIO26	
GPIO27	
GPIO14	
GPIO12	
GPIO13	
/RST	
3V3	
GND	
VBAT	
VUSB	
GND	
GPIO21	V_SPL_HD
GPIO1	U0_TXD
GPIO3	U0_RXD
GPIO22	V_SPL_WP
GPIO19	V_SPL_Q
GPIO23	V_SPL_D
GPIO18	V_SPL_CLK
GPIO5	V_SPL_CS0
GPIO15	ADC2_3
GPIO21	ADC2_2
GPIO0	ADC2_1
GPIO4	ADC2_0
GPIO17	U2_TXD
GPIO16	U2_RXD
3V3	
GND	
VBAT	
VUSB	
GND	
HSPI_CS0	Touch3
HSPI_WP	Touch2
HSPI_ID	Touch0
Button	

microB Connector
JST Connector for single celled LiPO
Power LED

Jumpers

Power
ESP32 VCC range: 2.2V-3.6V
VBAT: direct to battery (and charger)
VUSB: direct to USB (5V)
VCC: Output of regulator 3.3V/600mA
Up to 250mA during RF transmissions

Wireless
Wifi: 802.11 b/g/n/e/i
WPA/WPA2/WPA2-Enterprise/SPS
Bluetooth: Bluetooth 4.2/BLE

ESP32
Dual-core Xtensa 32-bit LX6
Up to 240MHz
520kB internal SRAM
4MB external flash

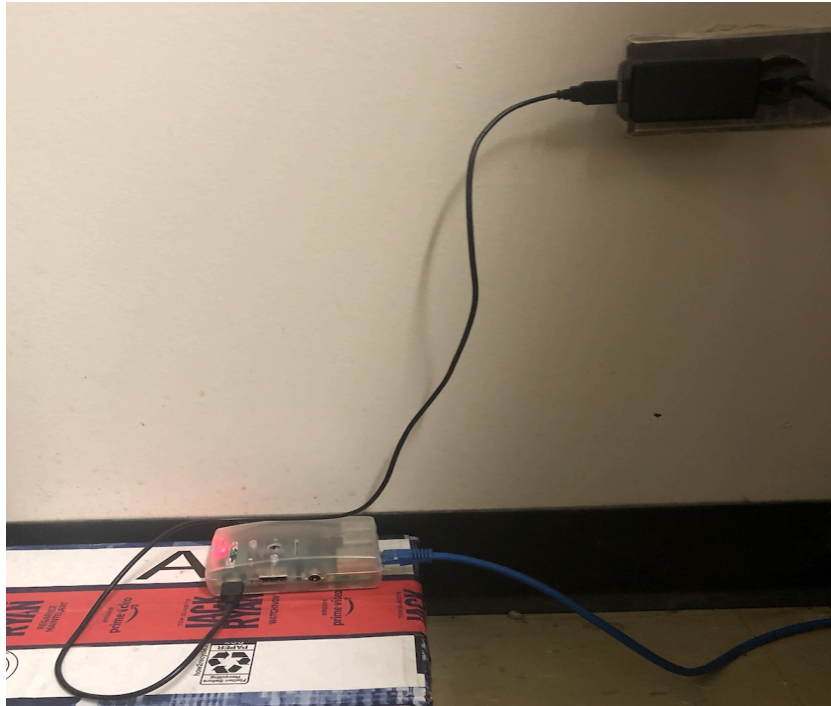
Multiplexed I/Os allow up to
18 ADC channels
3 SPI interfaces
3 UART interfaces
2 I2C interfaces
2 I2S interfaces
16 LED PWM outputs
2 DACs
10 Capacitive Touch Inputs

ADC Preamp
When using preamp 270pF cap
connector from VP to CAPP

Other
Hall Sensor
Temp sensor (-40C to 125C)
SD/SDIO/MMC Host controller

Raspberry Pi Single-Board Computer: MTU

- Hosted the Thinger.IO IoT Server
- Connected to MUN Network via an Ethernet cable in Configuration A.
- Connected to Wi-Fi Router LAN Port via an Ethernet cable in Configuration B.



SCADA Communication Channel: Wi-Fi Router for Private Wi-Fi Network

- D-Link Router (DI-524 Airplus G) used to create a private Wi-Fi Network

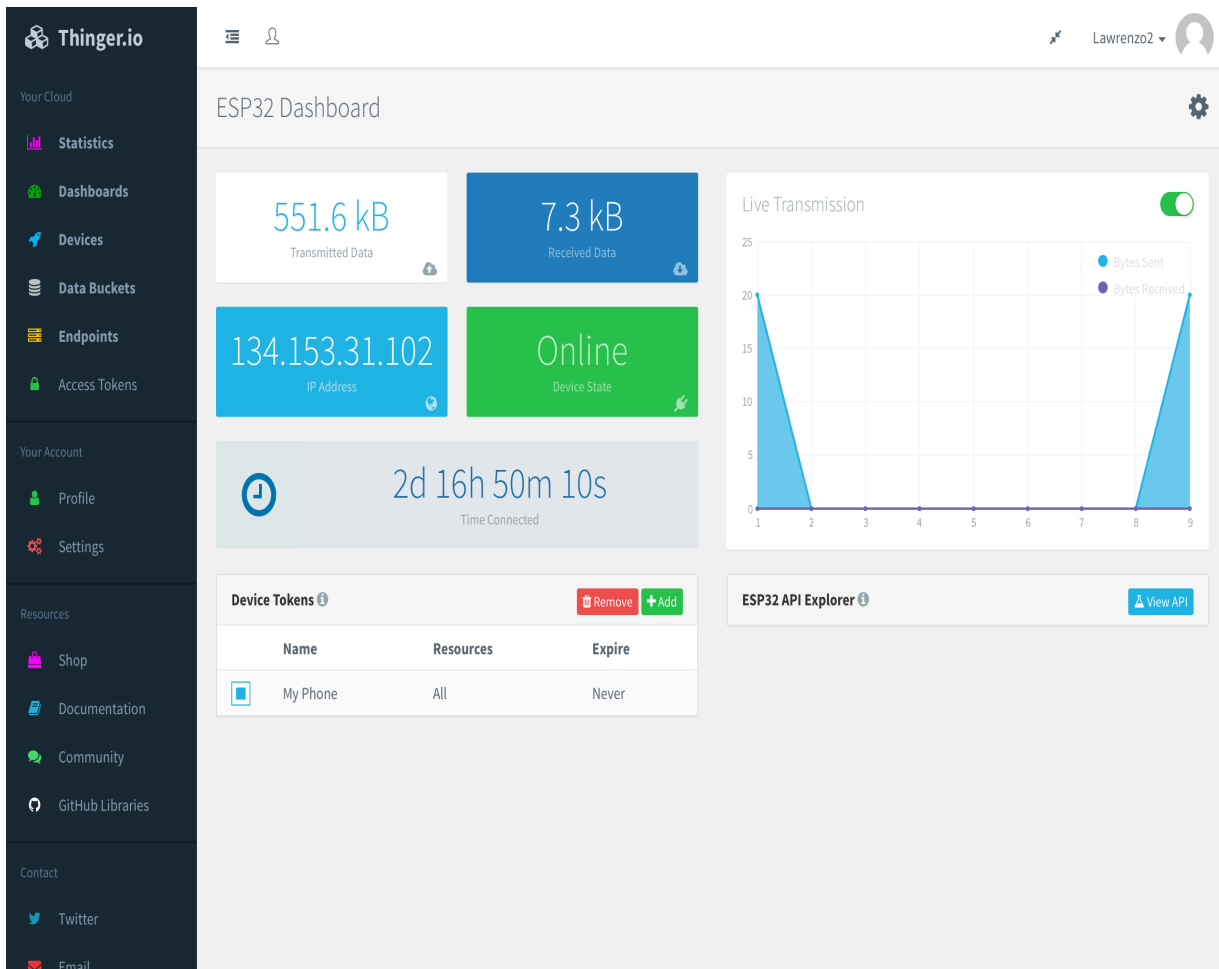


Thinger.IO IoT Server: Overview

- A Powerful Open Source platform for the Internet of Things; supported by GitHub
- Supports REST API for controlling & reading smart devices; e.g: sensors
- Uses HTTP methods like OPTIONS, GET, PUT, POST, and DELETE.
- Specific response of each resource gotten in JSON/XML format.
- Allows integration of Arduino-compatible hardware; e.g: ESP32
- Both Web-Server and Self-Hosted options available
- Connected Devices identified with Unique Identifier and Credentials
- Cloud Console shows Google Maps of Connected Devices
- Allows an operator to define 4 resources (Input, Output, Input/Output, Callback resources)
- Supports Real-time visualization dashboards & charts for remote monitoring
- Supervisory Controls possible using Endpoints (emails, HTTP Requests, etc)
- Free Android & iOS Apps for Server integration available

Thinger.IO IoT Server: Configurations

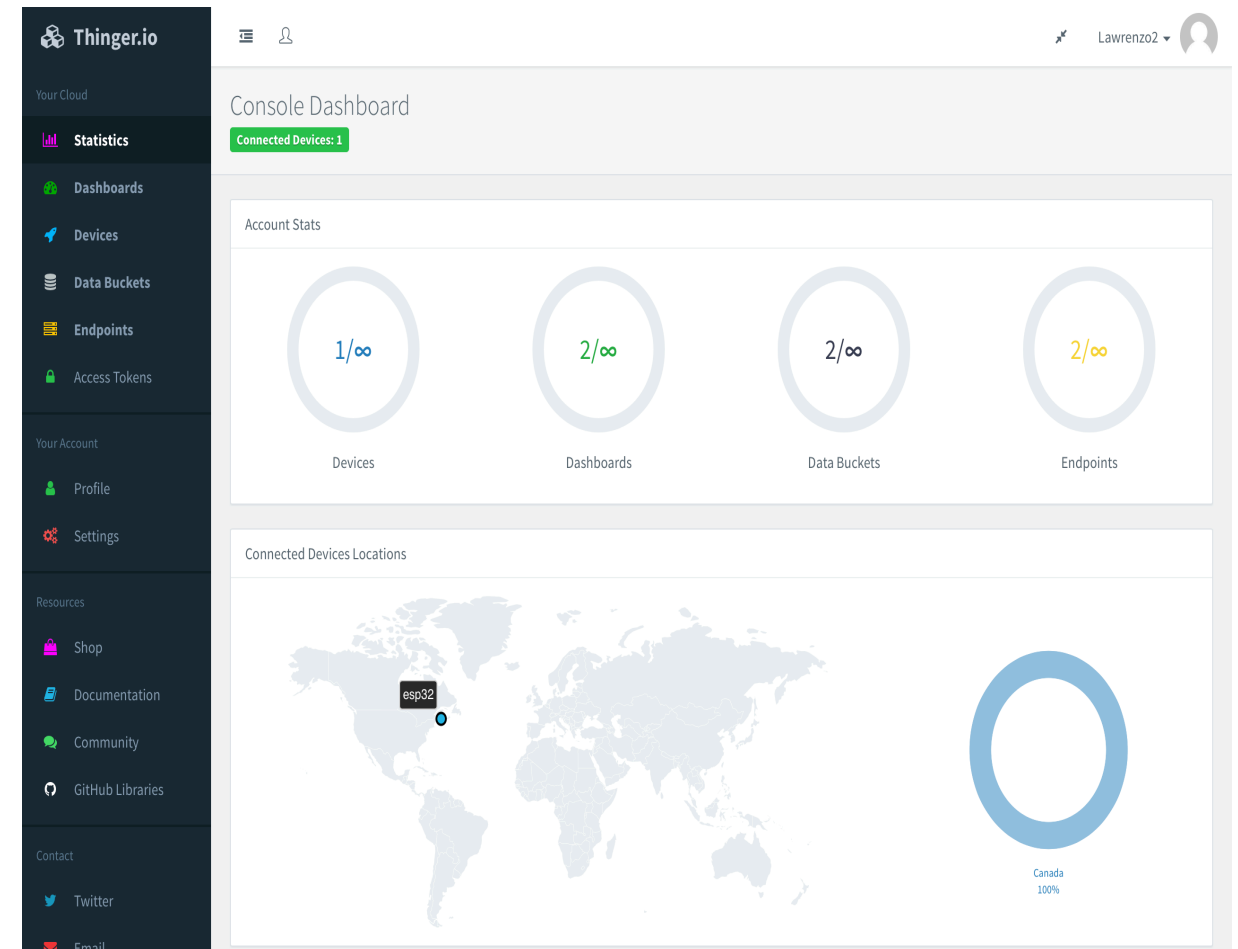
- Main Data Server installed on a Local Machine (Rpi), Self-hosted, Self-managed on private networks



The screenshot shows the Thinger.io ESP32 Dashboard. The interface includes a sidebar with navigation options like Statistics, Dashboards, Devices, Data Buckets, Endpoints, Access Tokens, Your Account (Profile, Settings), and Resources (Shop, Documentation, Community, GitHub Libraries, Contact). The main content area displays:

- ESP32 Dashboard title and user profile (Lawrenzo2).
- Summary cards: Transmitted Data (551.6 kB), Received Data (7.3 kB), IP Address (134.153.31.102), and Device State (Online).
- Time Connected: 2d 16h 50m 10s.
- Live Transmission graph showing Bytes Sent and Bytes Received over 9 data points.
- Device Tokens table with columns Name, Resources, and Expire.
- ESP32 API Explorer with a View API button.

Name	Resources	Expire
My Phone	All	Never

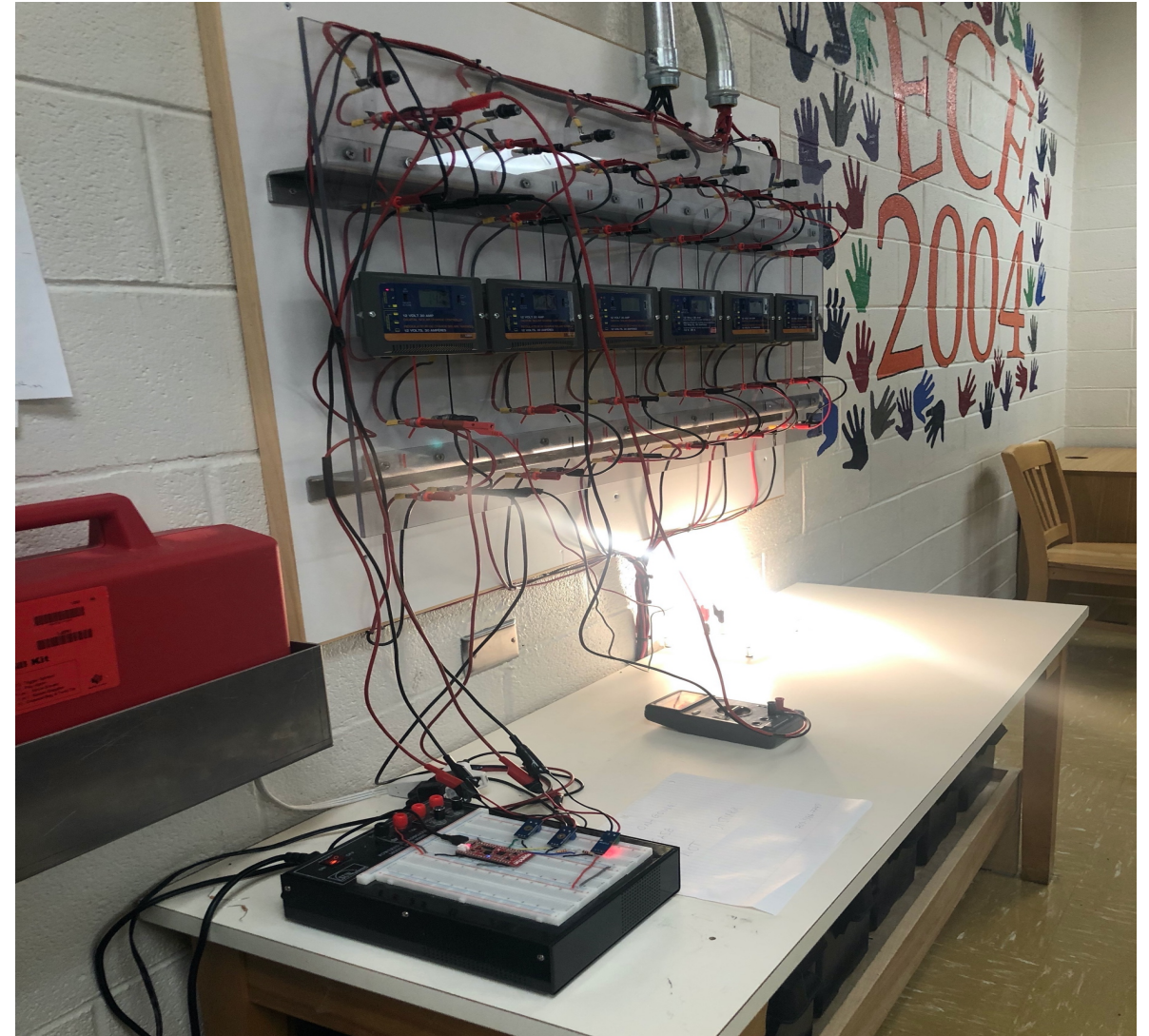
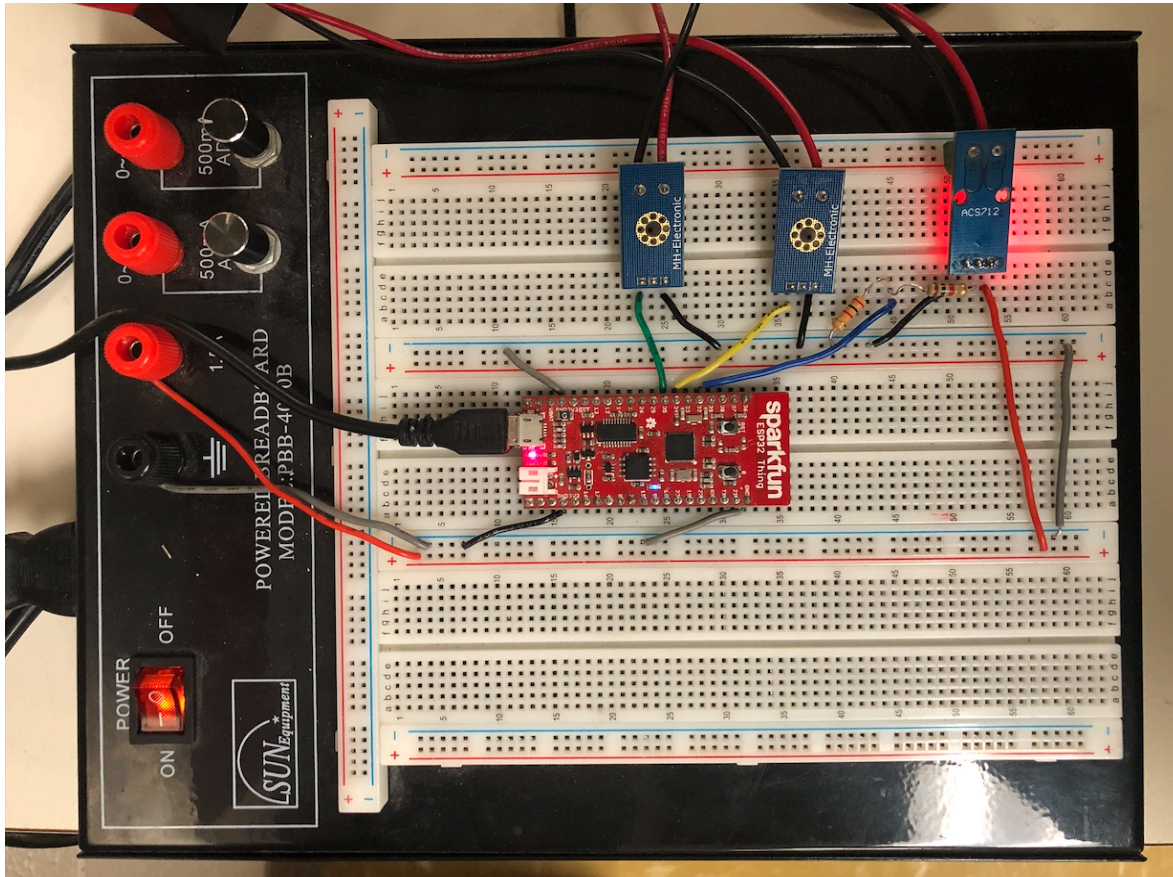


The screenshot shows the Thinger.io Console Dashboard. The interface includes a sidebar with navigation options like Statistics, Dashboards, Devices, Data Buckets, Endpoints, Access Tokens, Your Account (Profile, Settings), and Resources (Shop, Documentation, Community, GitHub Libraries, Contact). The main content area displays:

- Console Dashboard title and user profile (Lawrenzo2).
- Connected Devices: 1.
- Account Stats section with four circular gauges: Devices (1/∞), Dashboards (2/∞), Data Buckets (2/∞), and Endpoints (2/∞).
- Connected Devices Locations section with a world map showing a device location in North America and a donut chart for Canada (100%).

Prototype Design & Experimental Setup:

- Hardware implemented & System Set up on a Small Renewable Power Generation System (Solar PV System) at MUN ECE Lab:



Implementation Methodology & Flow Chart:

Algorithm 1: Data Logging Algorithm:

Initialization;

1. Read Sensor Values on Analog Pins 32, 34 and 35, and Calculate Values for Pins 32×34 ;
2. Display the above Values on Arduino IDE Serial Monitor;
3. Connect to Local Wi-Fi Network with Wi-Fi Name and Password;
4. Connect to Thinger.IO Local Server IP Address;
5. Identify the specified Thinger.IO Account Name, Device ID and Credentials;
6. Post Sensor Data to the specified Thinger.IO Device;

while Thinger.IO Server Acknowledges Data Receipt **do**

7. Display Sensor Data on Thinger.IO Cloud Console, and;
8. Display "Ok" on Arduino IDE Serial Monitor;

if No Data Receipt Acknowledgement from Thinger.IO Server **then**

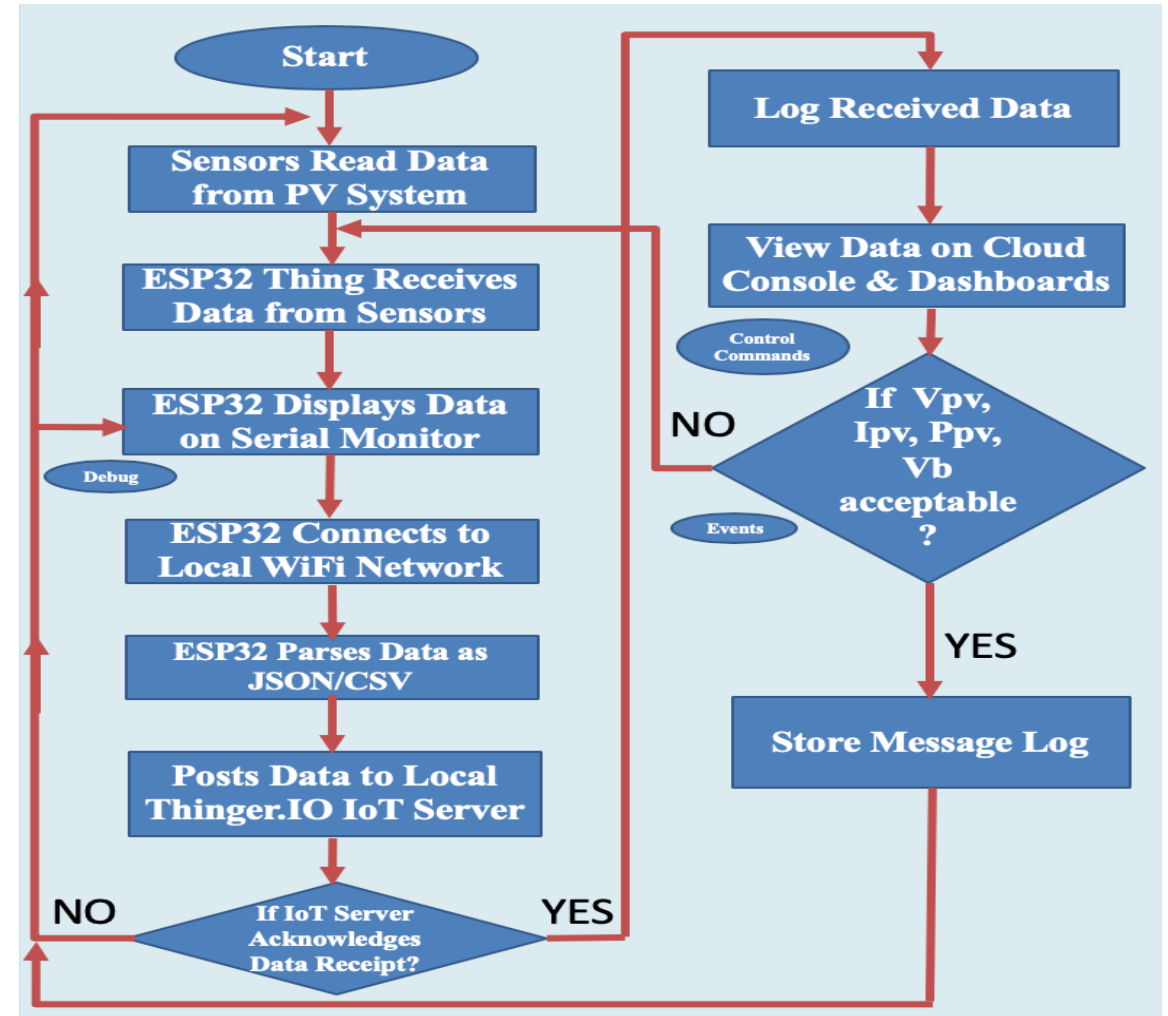
9. Display Debug/Error Message on Arduino IDE Serial Monitor;

else

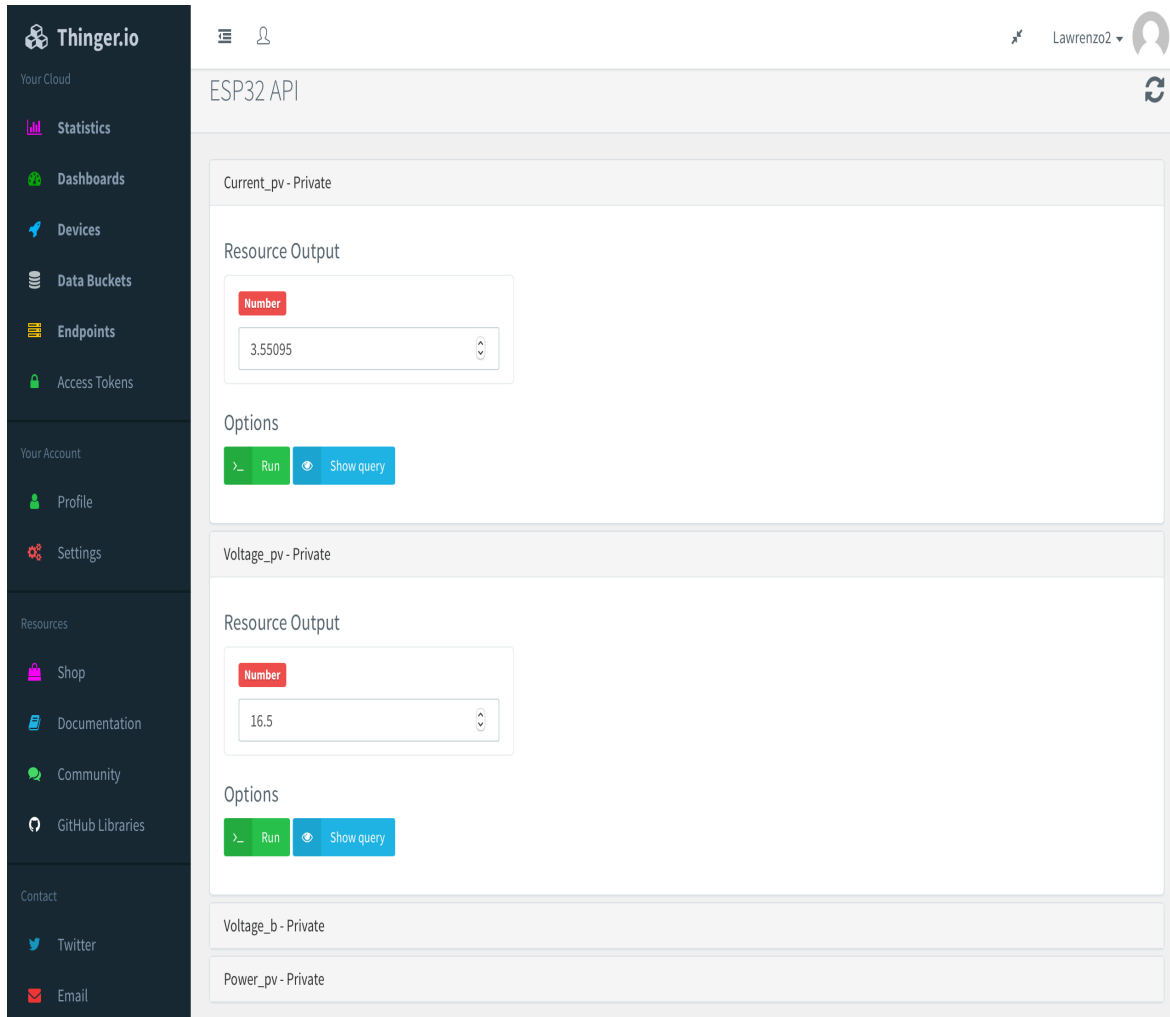
10. Go to Step 1;

end

end

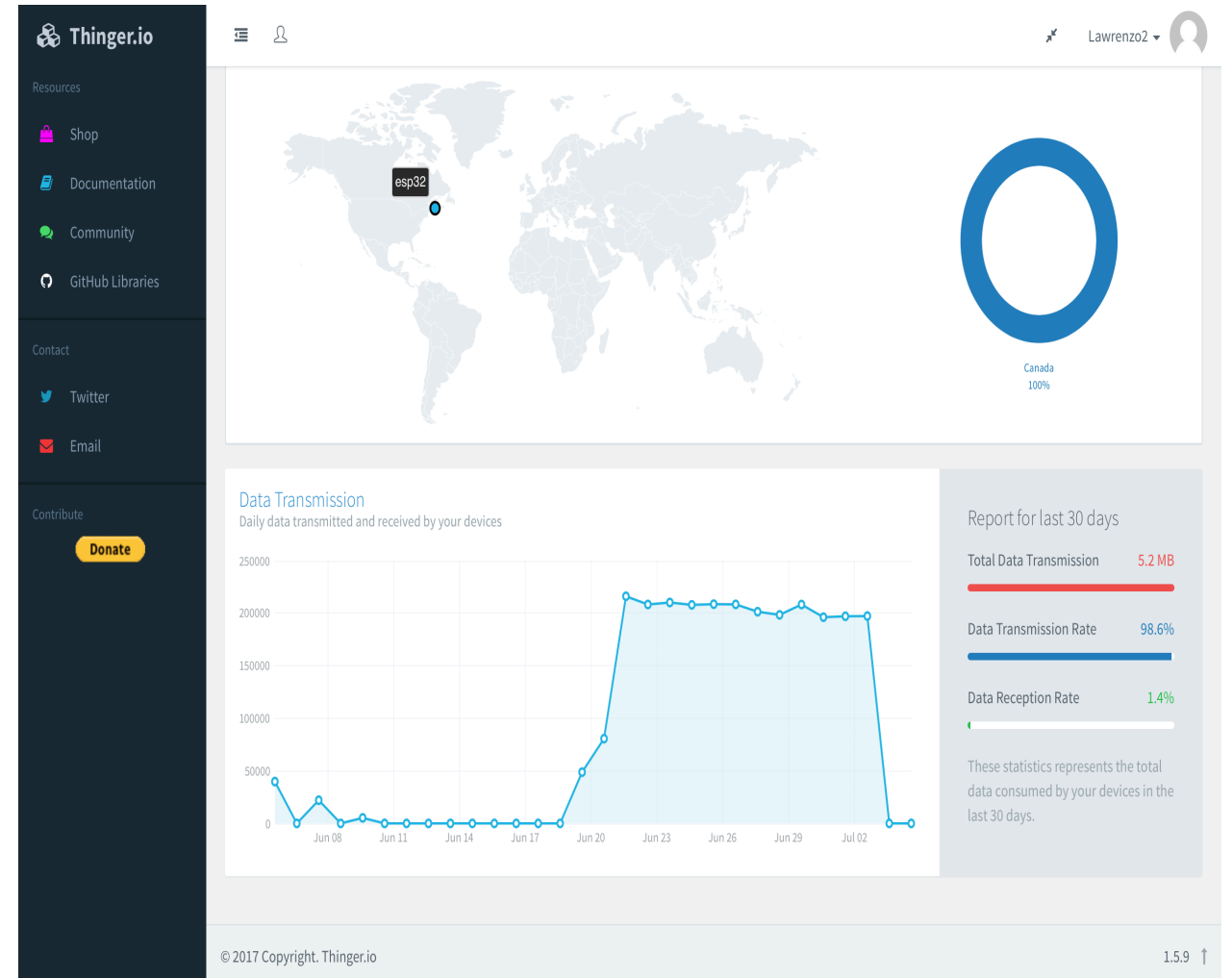


Data Logging & Historic Storage:



The screenshot shows the Thinger.io interface for an ESP32 API. The left sidebar contains navigation options: Your Cloud (Statistics, Dashboards, Devices, Data Buckets, Endpoints, Access Tokens), Your Account (Profile, Settings), Resources (Shop, Documentation, Community, GitHub Libraries), and Contact (Twitter, Email). The main content area displays three resource outputs:

- Current_pv - Private**: Resource Output is 3.55095. Options: Run, Show query.
- Voltage_pv - Private**: Resource Output is 16.5. Options: Run, Show query.
- Voltage_b - Private**
- Power_pv - Private**



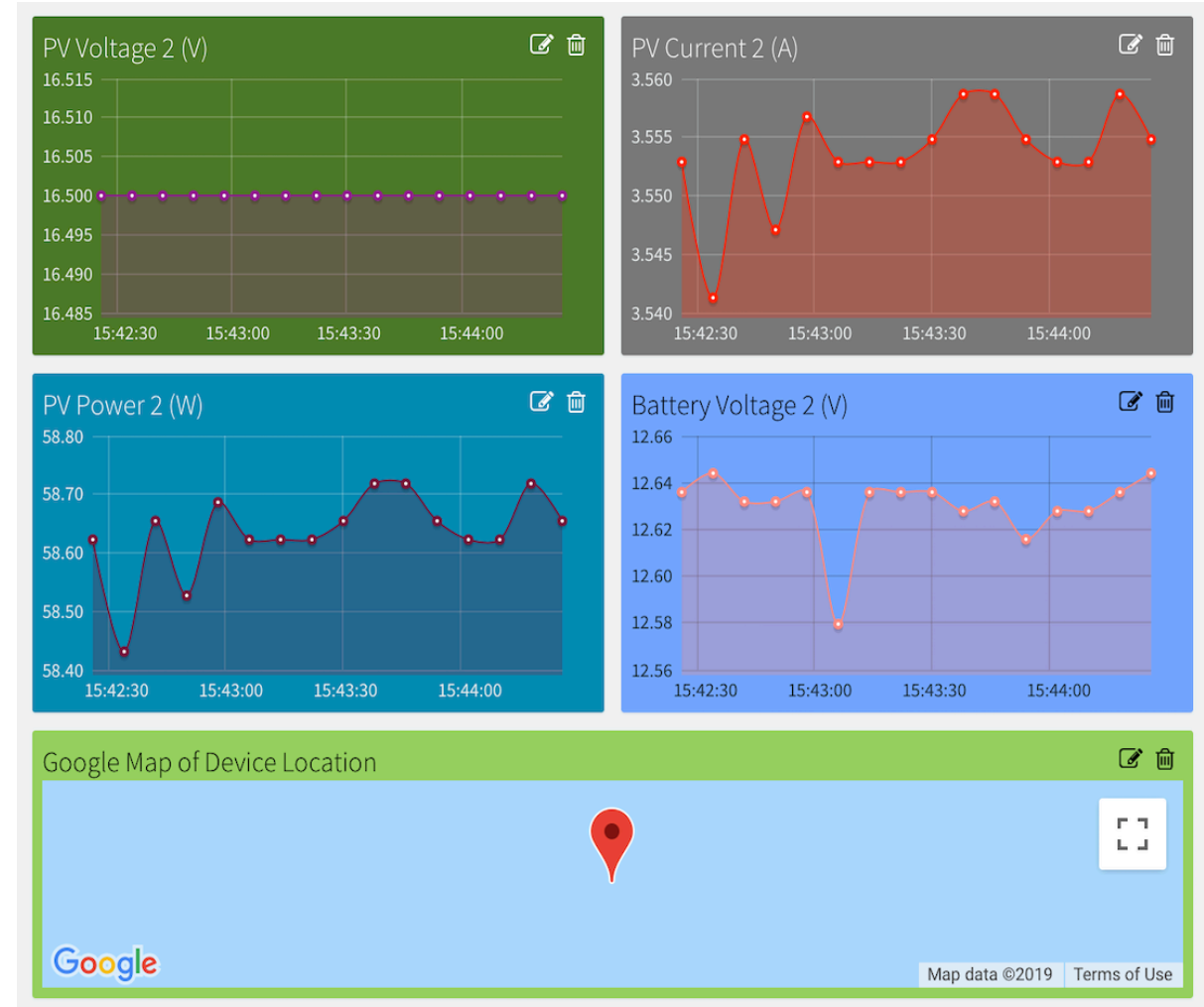
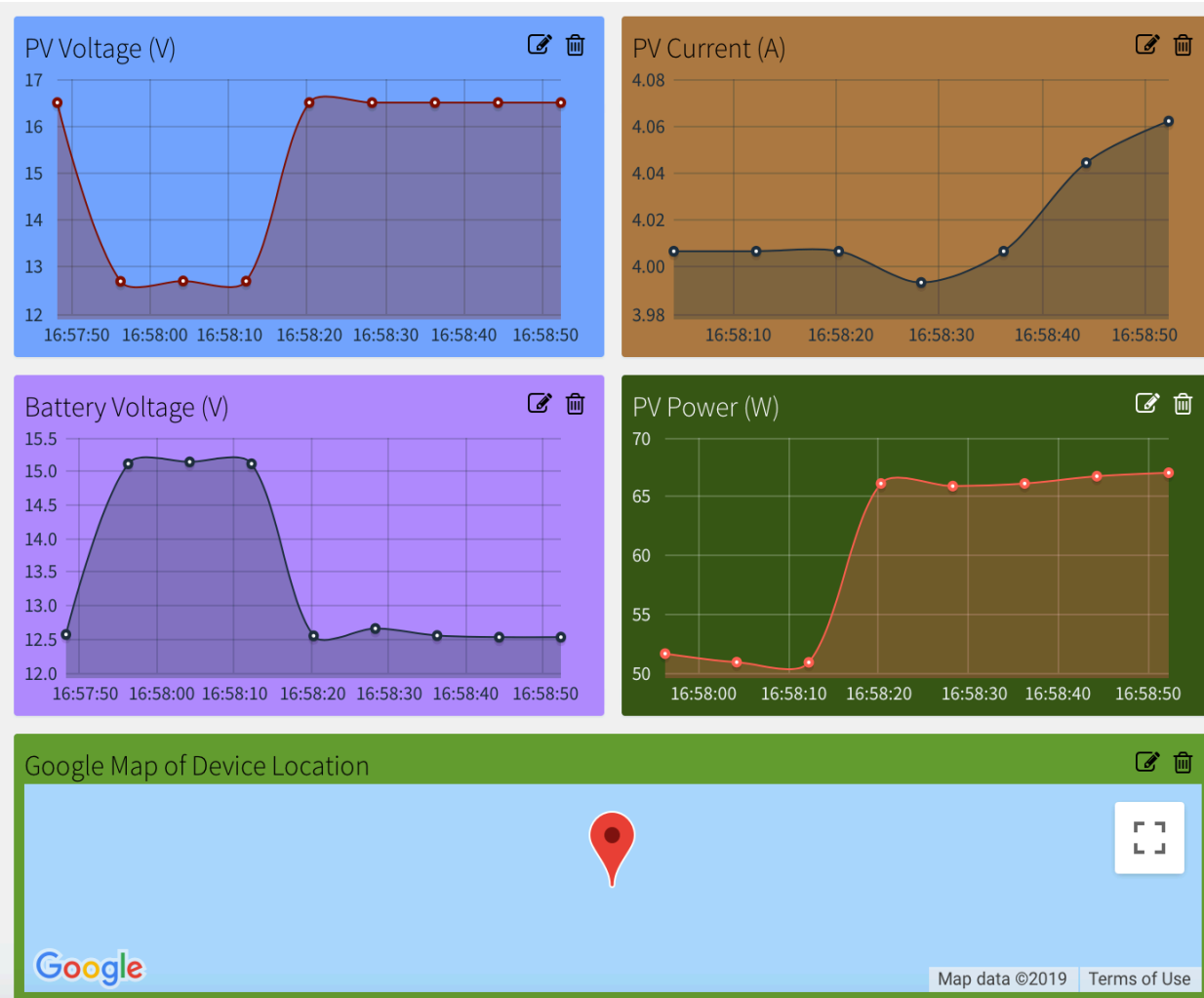
The screenshot shows the Thinger.io interface for data transmission. The left sidebar is identical to the previous screenshot. The main content area features a world map with a location marker for 'esp32' in Canada. To the right of the map is a donut chart showing 'Canada 100%'. Below the map is a 'Data Transmission' section with a line graph showing daily data transmitted and received from June 8 to July 2. The graph shows a significant spike in data transmission starting around June 20. To the right of the graph is a 'Report for last 30 days' summary:

Metric	Value
Total Data Transmission	5.2 MB
Data Transmission Rate	98.6%
Data Reception Rate	1.4%

These statistics represent the total data consumed by your devices in the last 30 days.

© 2017 Copyright. Thinger.io 1.5.9 ↑

Human Machine Interface (HMI): PV Data Monitoring & Supervisory Control:



Conclusions: Some Key Features of SCADA 2:

- IoT-based, Secure, Reliable, Low-cost, Low-Power, & Open Source
- Reporting, Remote Data Monitoring & Supervisory Control, Data Acquisition, Historic Data Storage & Trending, etc.

Table 4.1: Bill of Materials.

S/N	COMPONENT	QTY	PRICE (CAD)
1	Thinger.IO RPi ISO Image	1	15.62
2	Raspberry Pi 2 B	1	45.95
3	ESP32 Thing	1	31.90
4	Current Sensor	1	5.25
5	Voltage Sensor	2	11.98
6	D-Link D1-524 Wireless Router	1	98.51
7	8GB SD Card	1	12.66
8	Miscellaneous (Breadboard, Resistors, Wires, Boxes, etc.)	1	70.00
Grand Total:			\$ 291.87 CAD

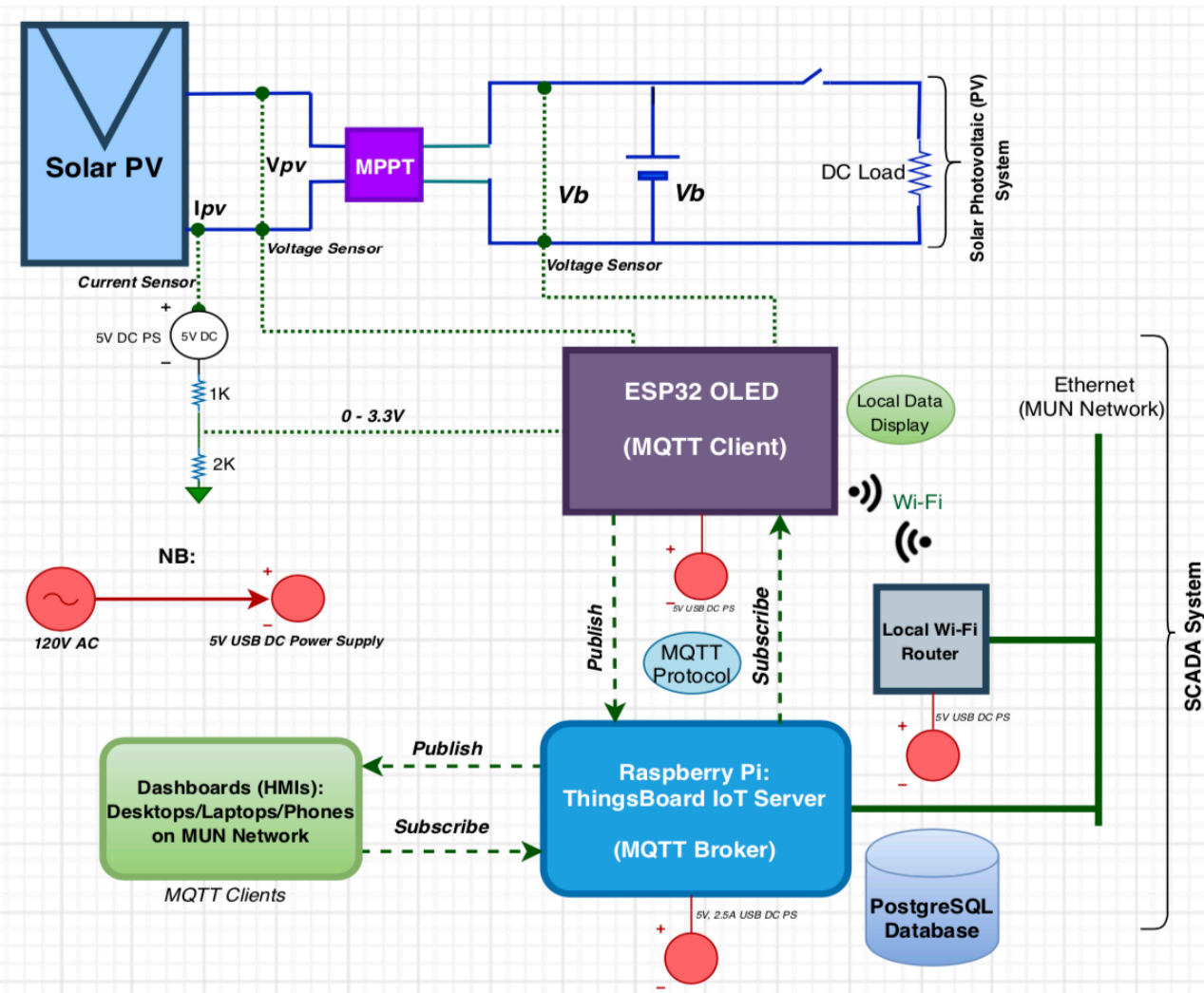
Table 4.2: Power consumption of hardware components.

S/N	HARDWARE	POWER (W)
1	Raspberry Pi 2 B	1.7
2	ESP32 Thing (alone)	0.5
3	Breadboard (with Sensors, ESP32, Resistors, etc. connected)	3.3
4	D-Link D1-524 Wireless Router	4.4
Total Power Consumption (less ESP32 alone):		9.4 W

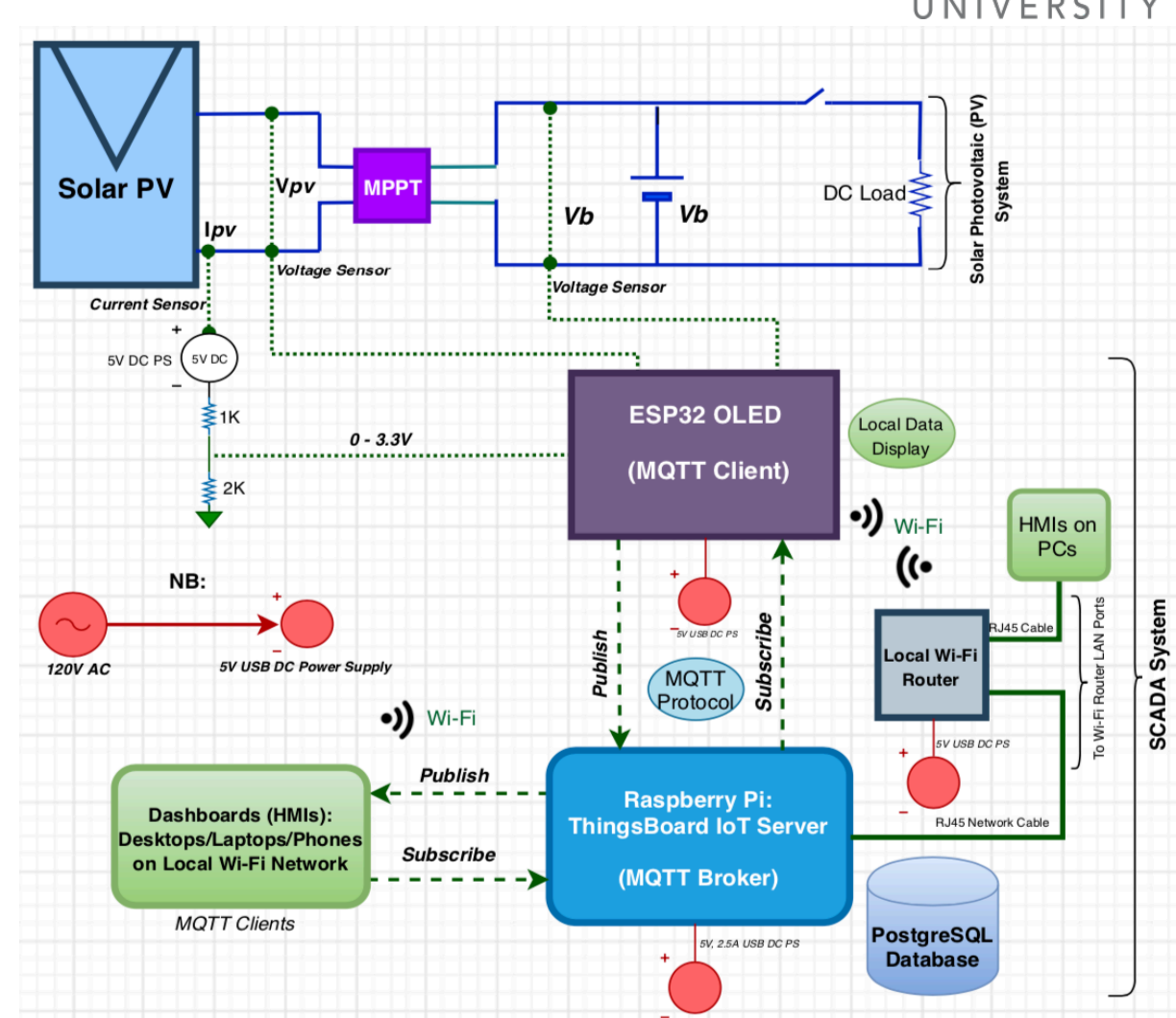
Components of SCADA 3 System:

- A Small Renewable Power Generation System (Solar PV System with ESS).
- Sensors (Current, Voltage.): FIDs
- ESP32 with OLED Micro-controller: RTU
- Wi-Fi Router: SCADA Communication Network
- MQTT Protocol: Data Transfer
- Raspberry Pi2 machine (ThingsBoard IoT Server): MTU
- Computers and Mobile Devices: HMIs/GUIs

System Configurations; Two Configurations considered



Configuration A: MUN Network (Internet Required)

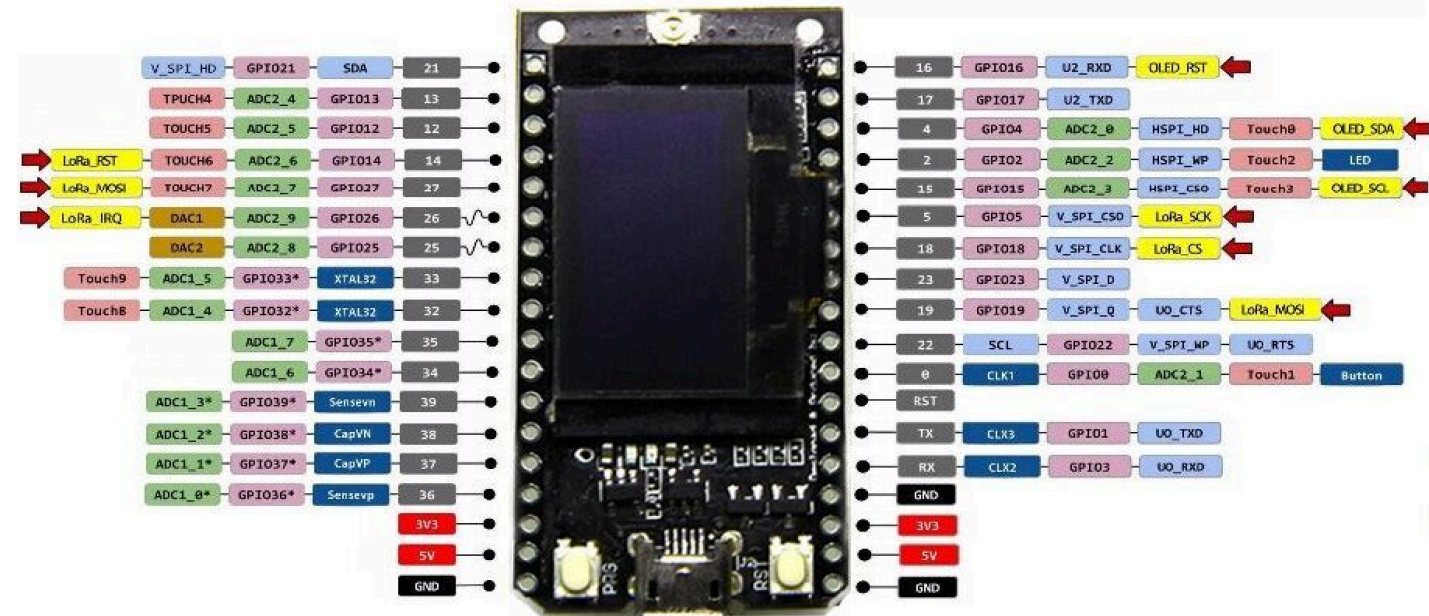
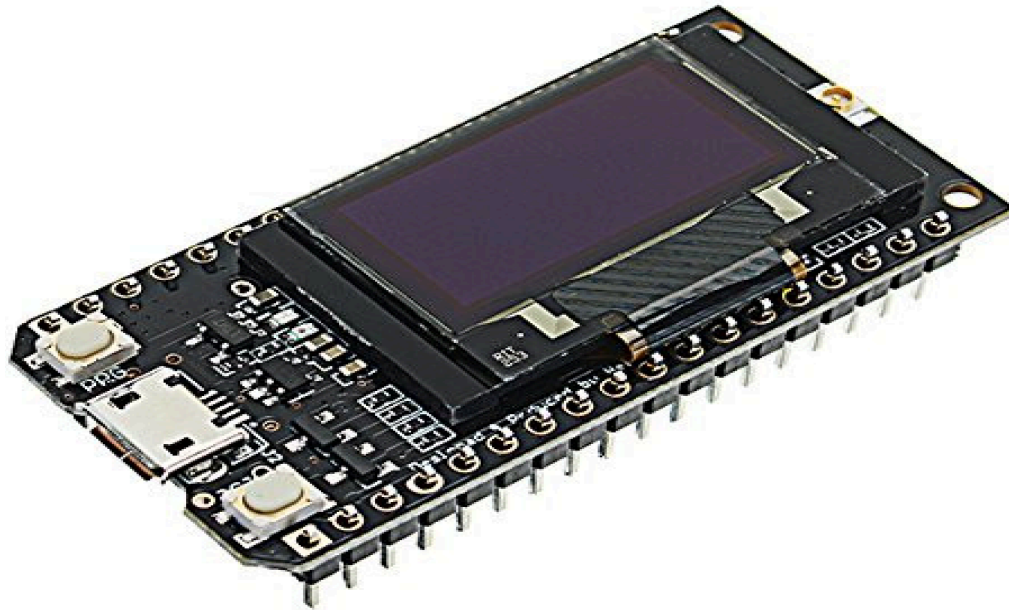
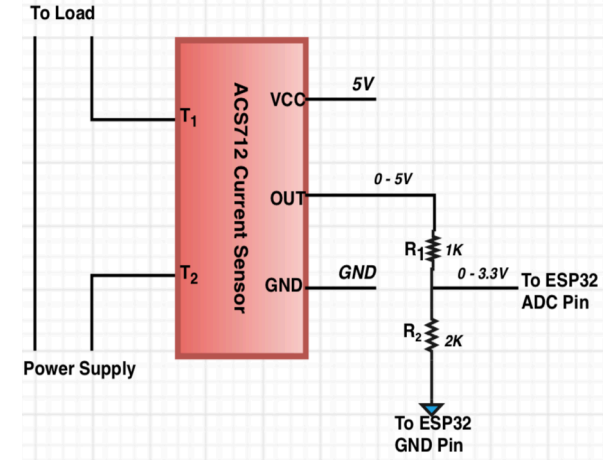


Configuration B: Private Industrial Network created (Internet Optional)

Field Instrumentation Devices: Same Sensors as in SCADA 1

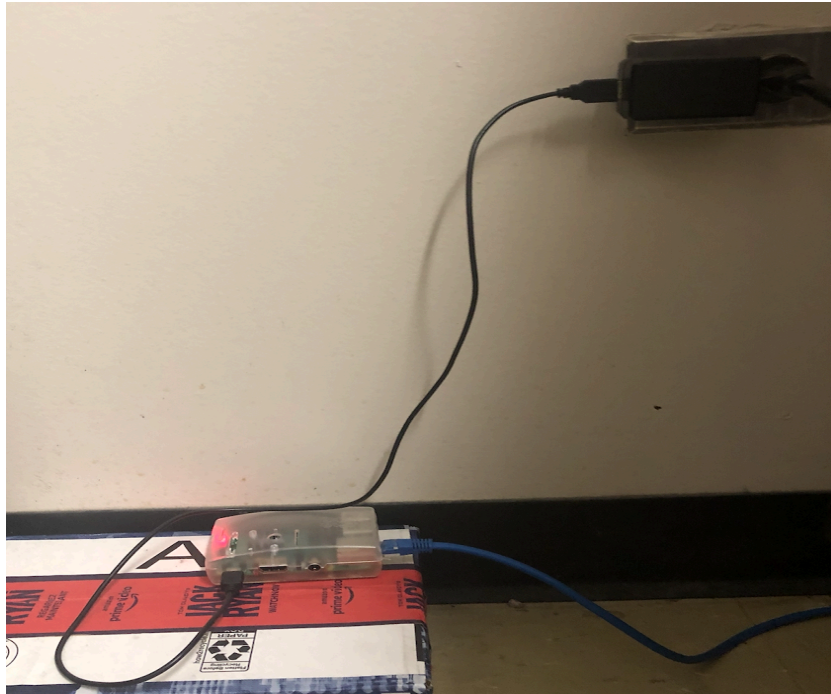
ESP32 with OLED: RTU

- TTGO ESP32 LoRa32 micro-controller board with 0.96 inch OLED display
- 18 ADC pins and over 30 GPIO pins (I/O Pins), Supports Wi-Fi
- Low-Cost (\$20), Low-Power (0.9W) Micro-controller for IoT Projects
- 5 V Power Supply, 1.8 V to 3.7 V ((3.3 V) ADC Signal Voltage
- Programmed as an MQTT Client using Arduino IDE & MQTT Client Library (PubSubClient).



Raspberry Pi Single-Board Computer: MTU

- ThingsBoard IoT Server & PostgreSQL Database installed on the Raspberry Pi
- Connected to MUN Network via an Ethernet cable in Configuration A.
- Connected to Wi-Fi Router LAN Port via an Ethernet cable in Configuration B.



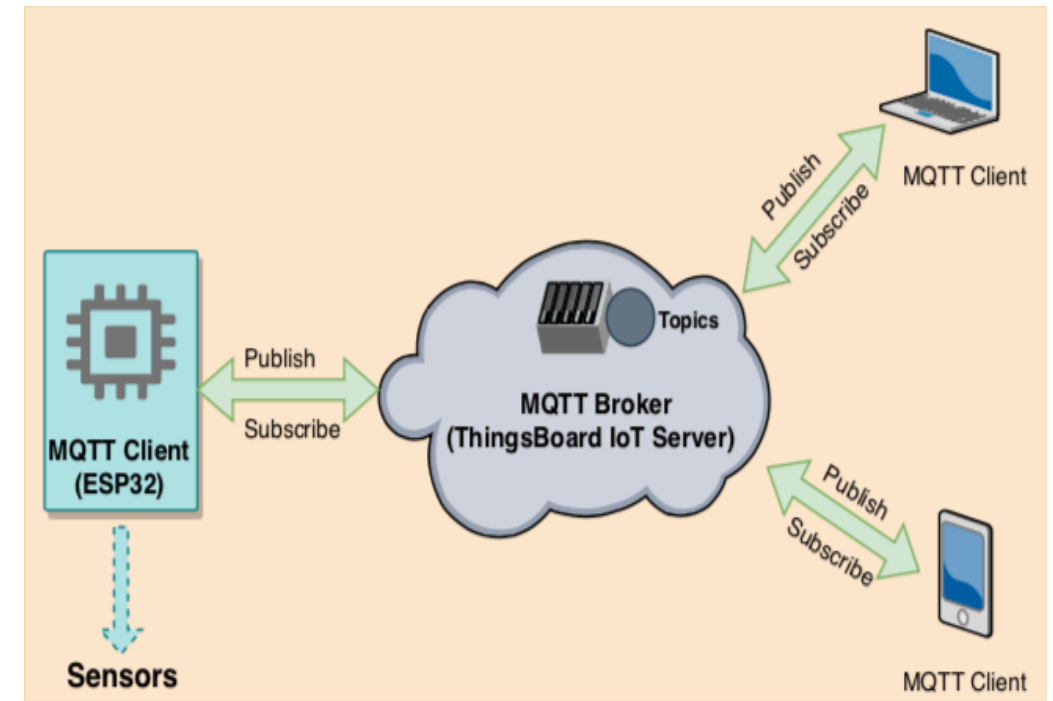
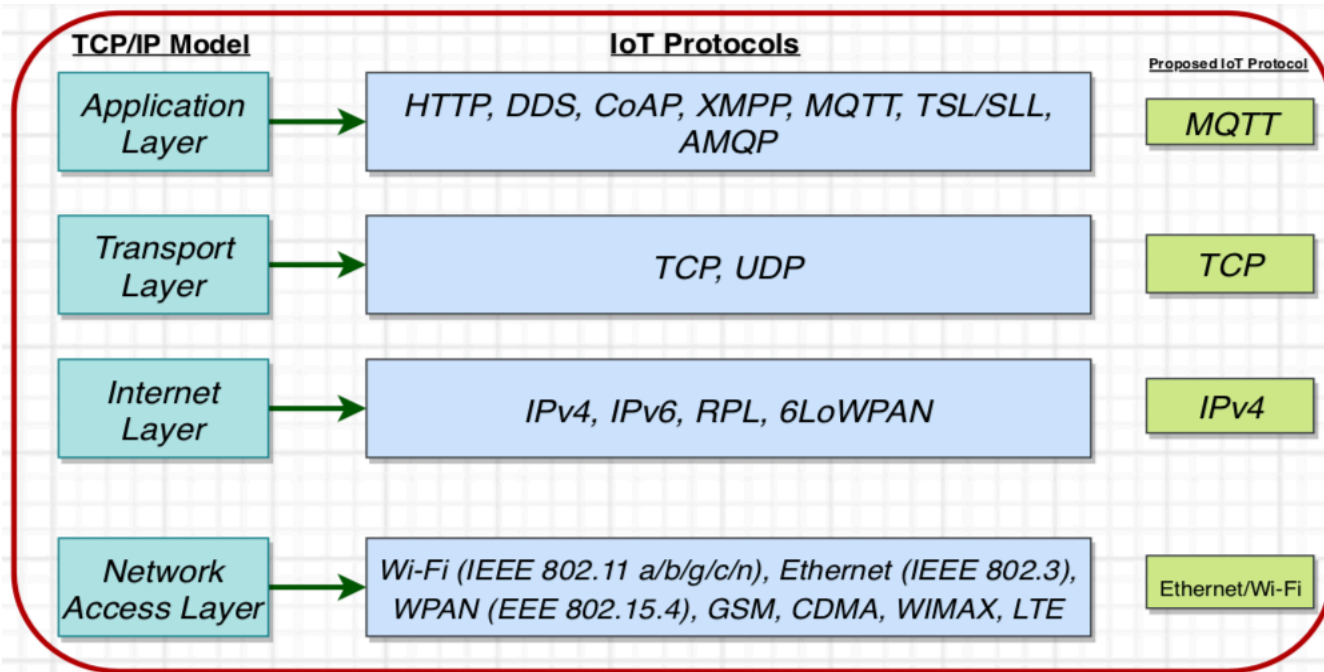
SCADA Communication Channel: Wi-Fi Router for Private Wi-Fi Network:

- D-Link Router (DI-524 Airplus G) used to create a private Wi-Fi Network



Internet of Things (IoT) & MQTT Protocol: Overview

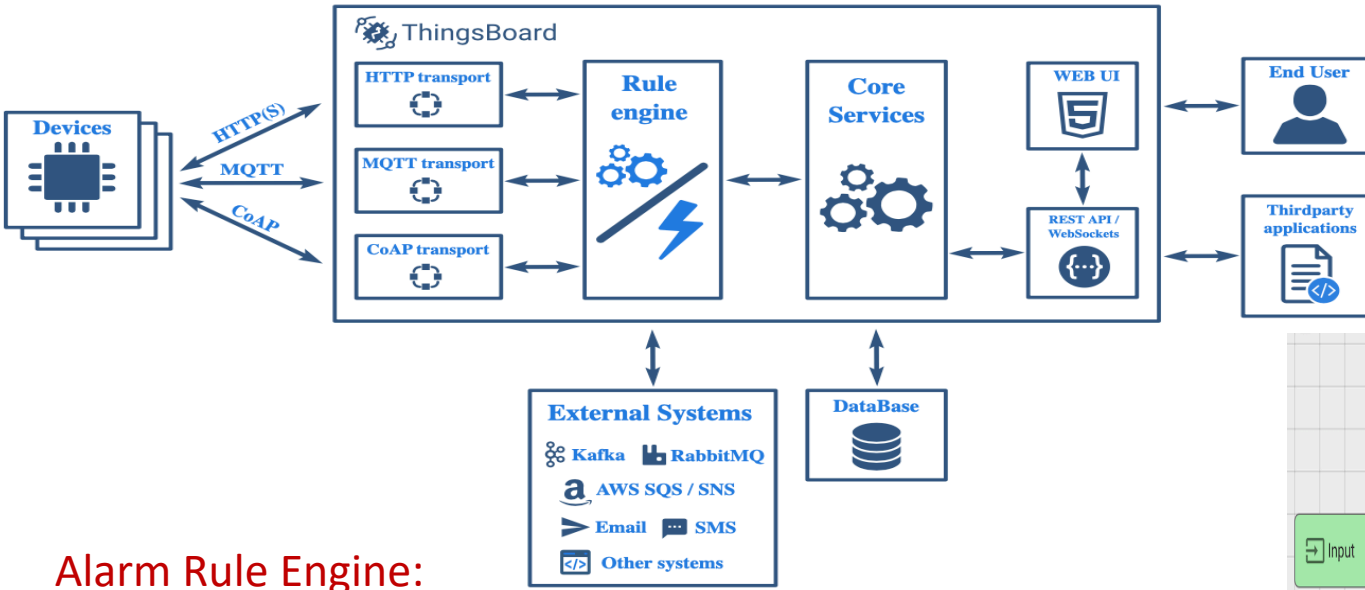
- A reliable communication is necessary in an IoT-based application like SCADA
- IoT Technologies use the four layers of the general TCP/IP model.
- Important choice for IoT-based SCADA: communication protocols, message encoding format, & the web/IoT platform
- MQTT: a lightweight M2M data communication protocol; supports applications with limited resources, e.g low bandwidth
- MQTT Protocol Implemented over TCP/IP Wi-Fi for sensor data transfer.



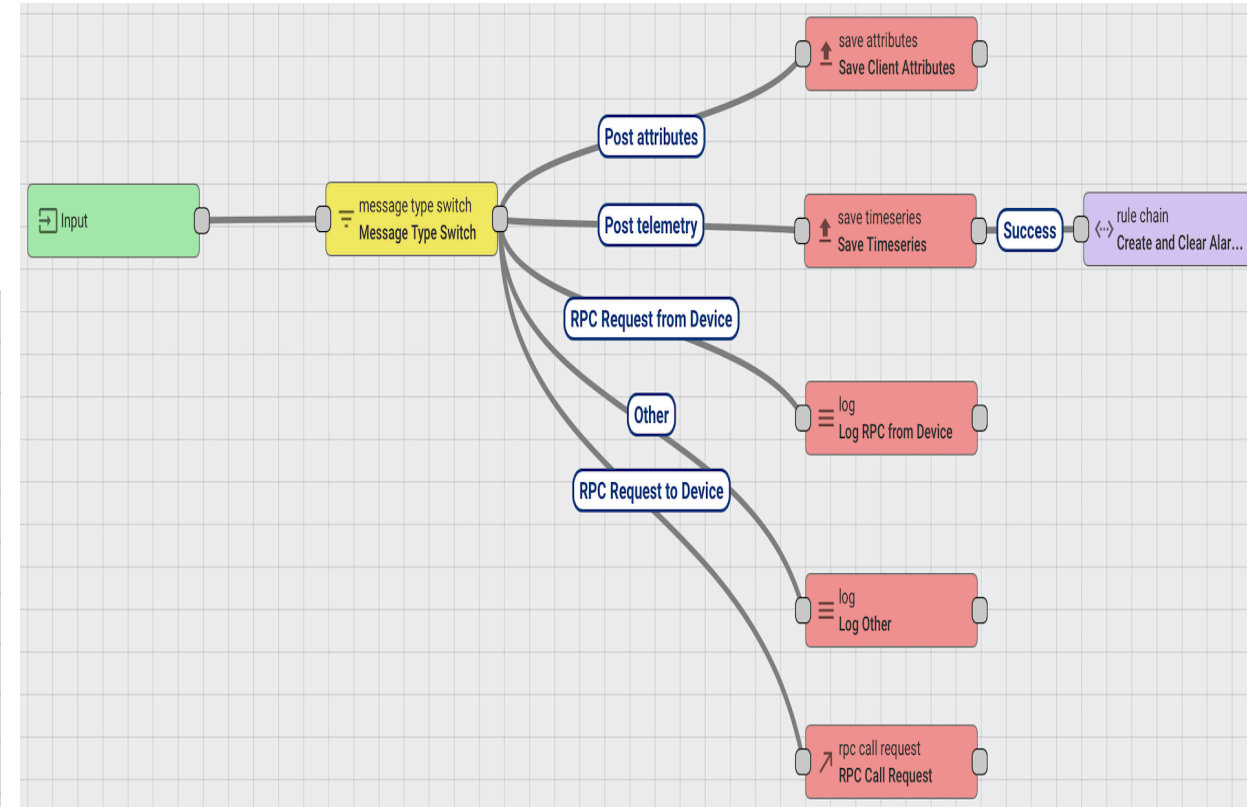
ThingsBoard IoT Server: Overview

- An Open Source IoT Platform for data collection, processing, visualization & device management
- 100% Support for standard IoT protocols, e.g.; MQTT, CoAP, etc.
- Live Demo (Web), Cloud Server (e.g.; AWS) & On-premise Server options available.
- Supports various hardware integration, e.g.; ESP32, Arduino, ESP8266, etc.
- Supports customizable real-time dashboards for data visualization, alarms, device management, etc.
- Server nodes act as MQTT Broker that supports QoS levels 0 & 1
- Uses Database to store Entities (e.g.; devices, dashboards) & Telemetry Data (e.g.; Sensor Readings)
- Supports 3 different Database options; SQL (e.g.; Postgres), NoSQL (e.g.; Cassandra), & Hybrid
- Stores received values as Telemetries: Time-series key-value pairs.
- Community & Professional Editions available: Community Edition used (Free & supported by GitHub)
- On-premise Server option installed on a Raspberry Pi alongside PostgreSQL Database: Data Security strategy.

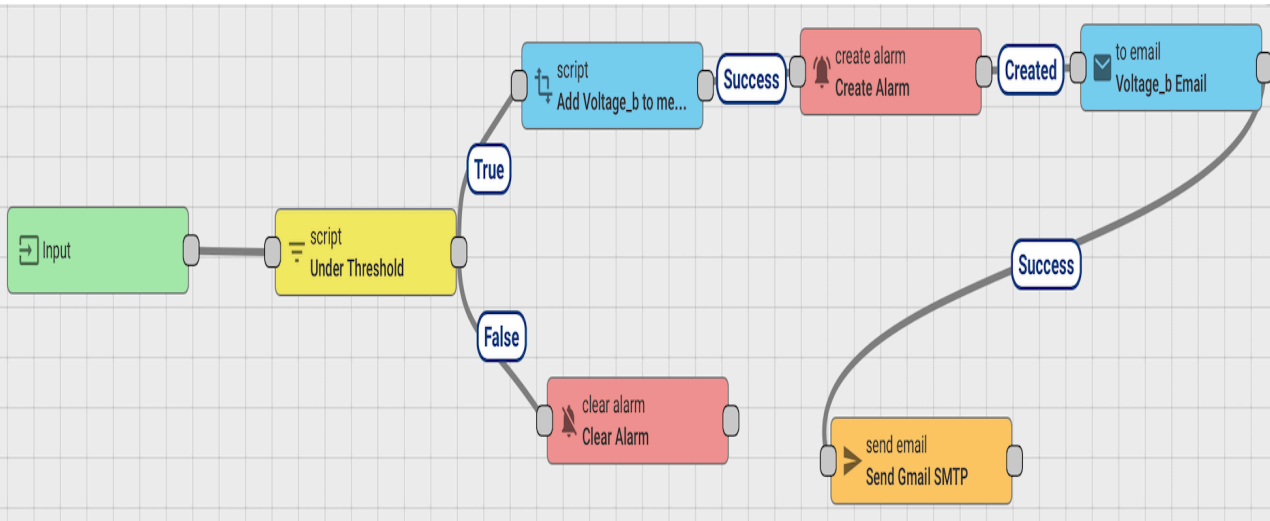
ThingsBoard IoT Server: Architecture



System Rule Engine:

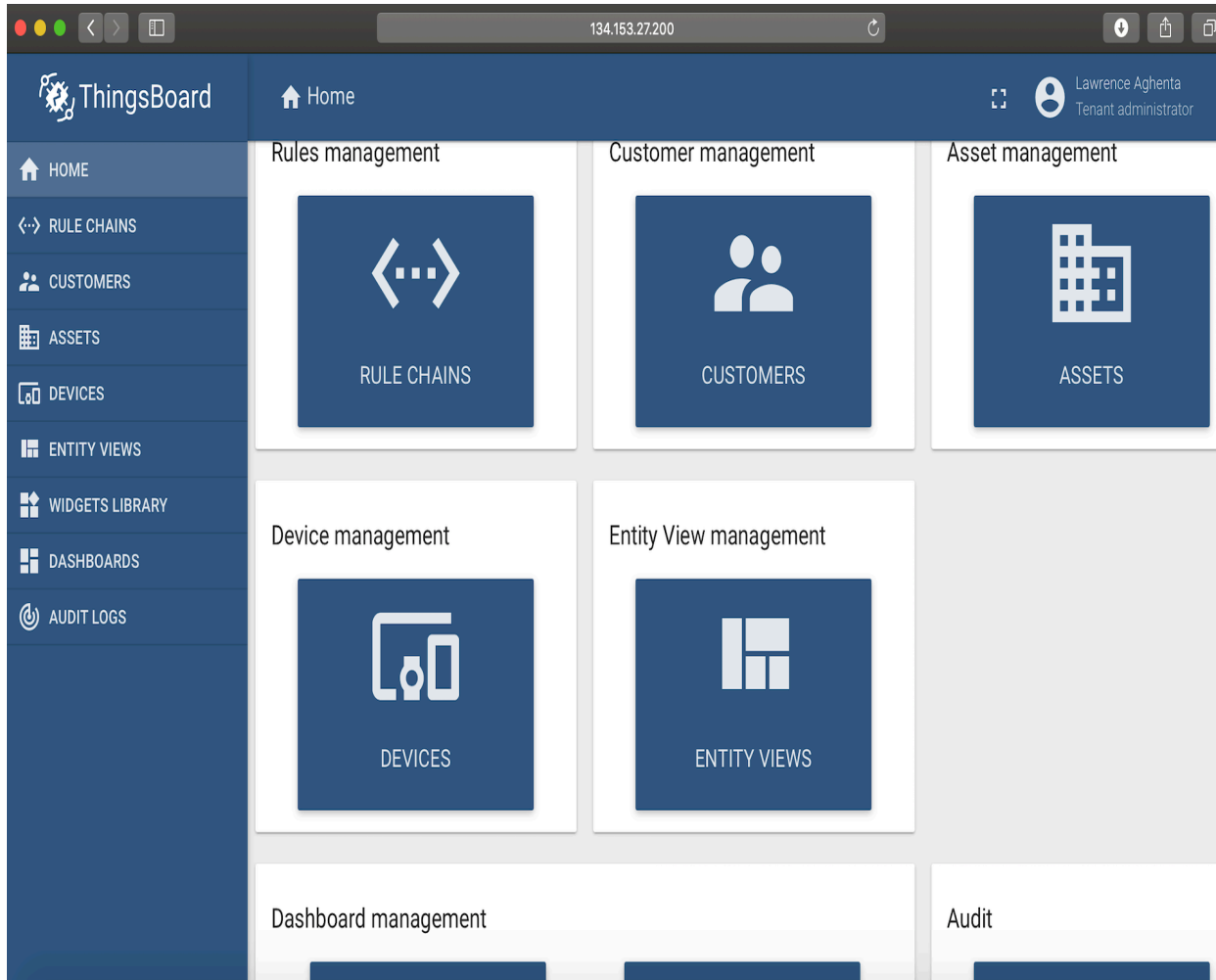


Alarm Rule Engine:

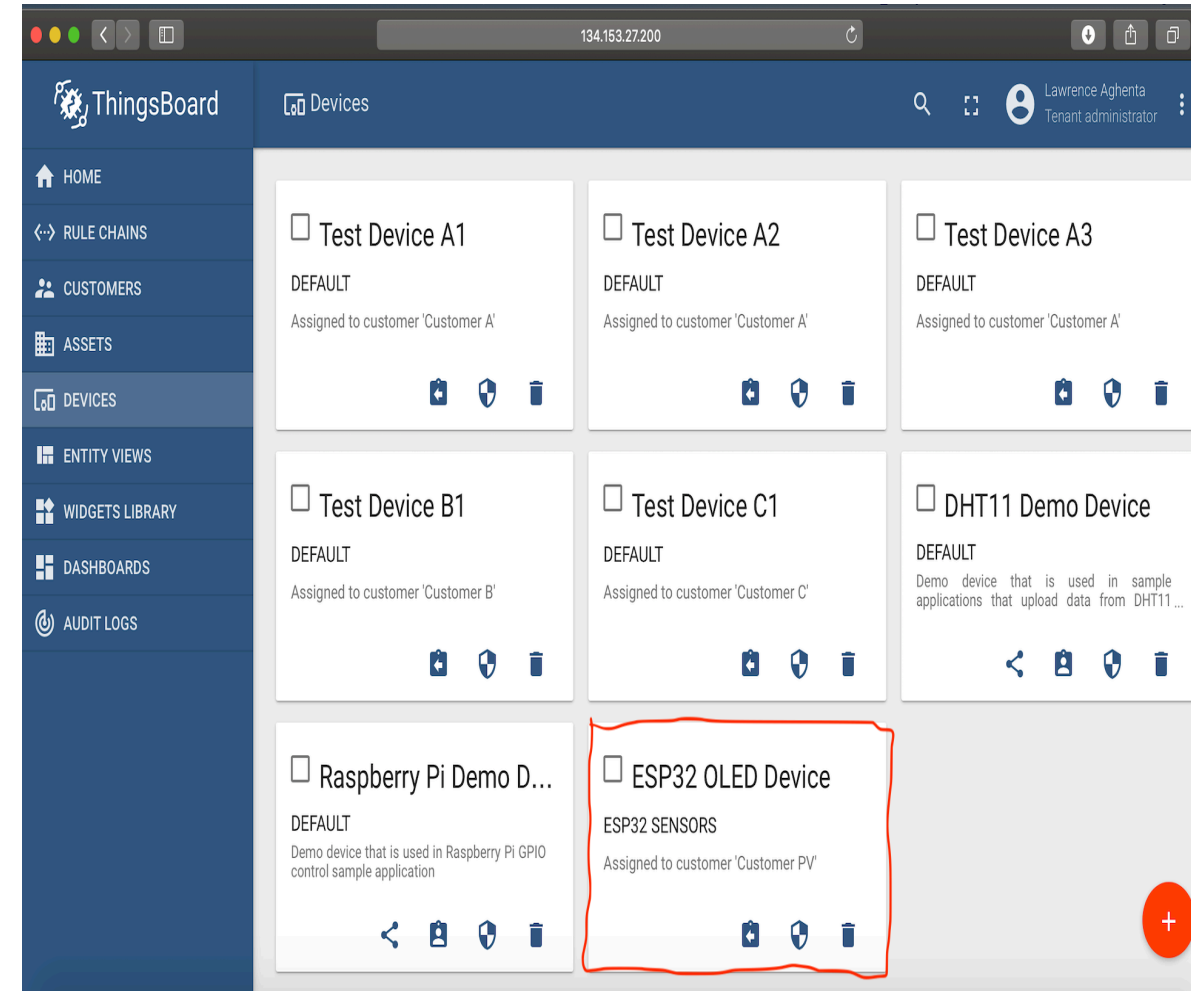


ThingsBoard IoT Server: Local Server Configuration

Rpi-installed ThingsBoard Server Platform:



Connected ESP32 OLED Device:



Implementation Methodology:

Algorithm 2: Data acquisition and logging algorithm:

Initialization;

1. Analog sensors measure and collect PV system data;
2. ESP32 reads sensor values on analog Pins 32, 34 and 35, and calculates values for Pins 32×34;
3. ESP32 displays the above values on Arduino IDE Serial Monitor and ESP32 OLED Screen;
4. ESP32 connects to local TCP/IP Wi-Fi Network with Wi-Fi Name and Password;
5. ESP32 MQTT Client identifies the local ThingsBoard IoT Server (MQTT Broker) via the Server IP Address;
6. ESP32 MQTT Client publishes sensor data to MQTT Broker over the TCP/IP Wi-Fi connectivity;
7. ThingsBoard Server displays data as Telemetry Messages on the specified Device using the Device Name and Access Token;
8. ThingsBoard Server Node logs the Telemetry Messages to Dashboards for data visualization;

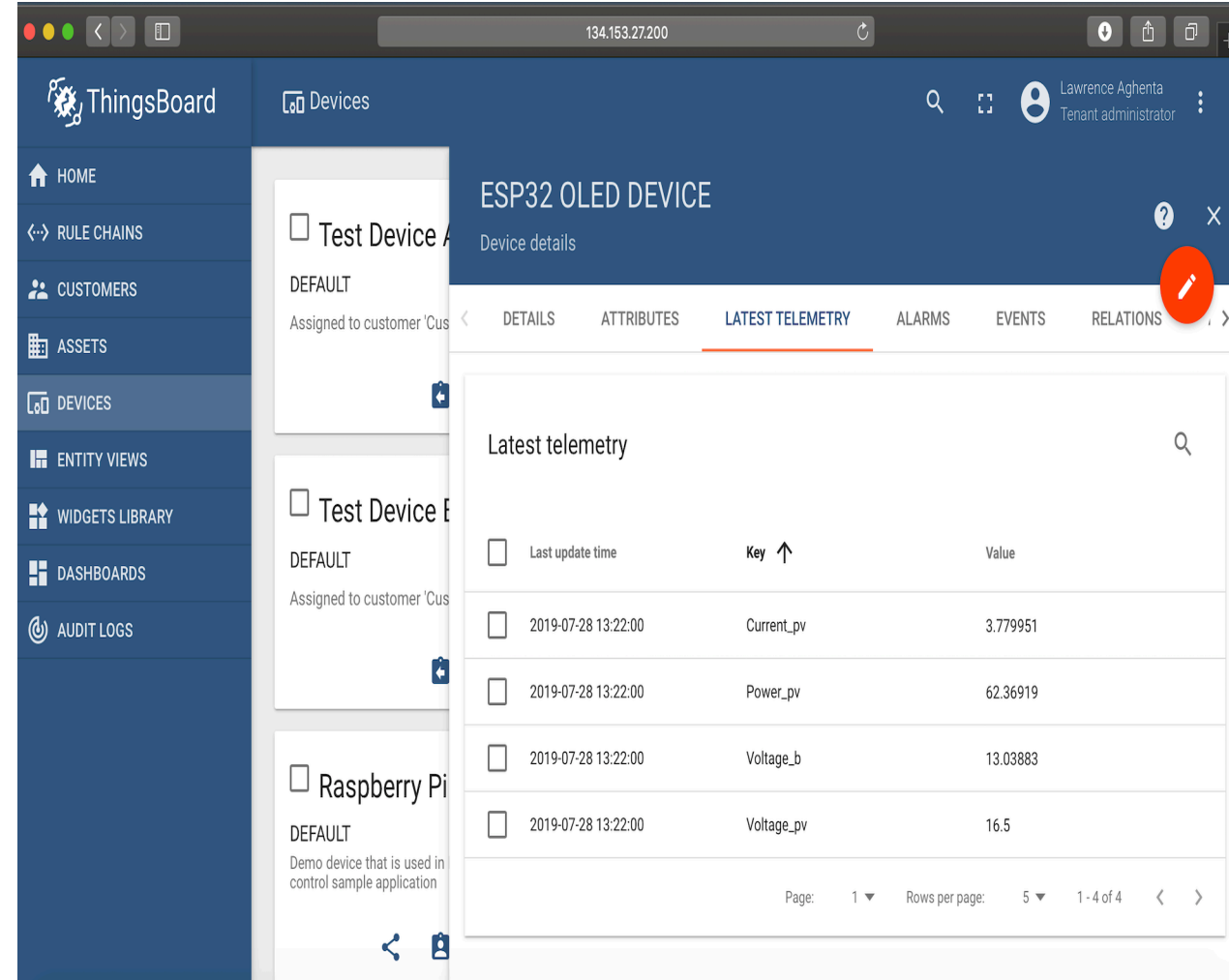
while ThingsBoard Server acknowledges data receipt do

9. Display sensor data on ThingsBoard Server Node, Dashboards and ESP32 OLED Screen, and;
10. Display "DONE" on Arduino IDE Serial Monitor;
- if No data receipt acknowledgement from ThingsBoard Server Node then
 11. Display "FAILED.....retrying in 5 seconds" on Arduino IDE Serial Monitor;
- else
 12. Go to step 1;

end

end

Real-time PV Data Posting:



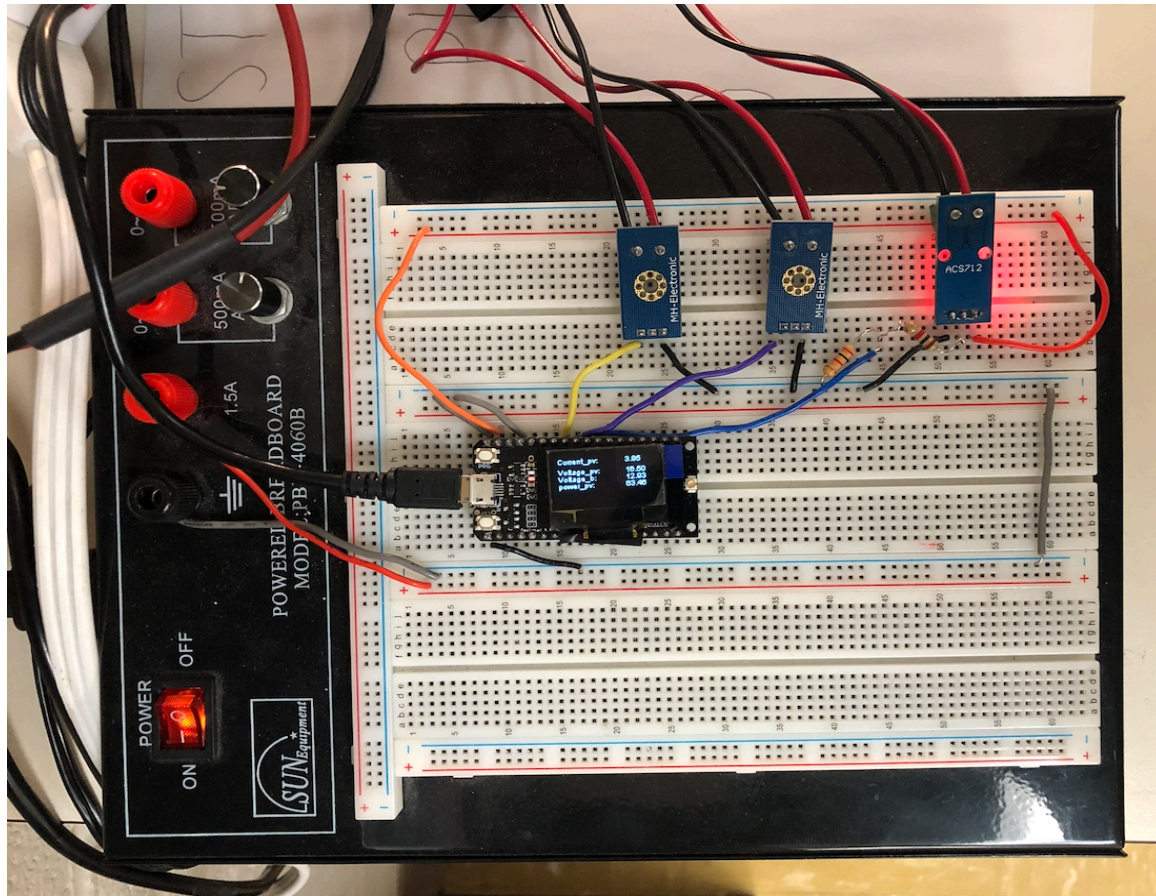
The screenshot shows the ThingsBoard web interface. The left sidebar contains navigation options: HOME, RULE CHAINS, CUSTOMERS, ASSETS, DEVICES, ENTITY VIEWS, WIDGETS LIBRARY, DASHBOARDS, and AUDIT LOGS. The main content area displays the 'ESP32 OLED DEVICE' details. Under the 'LATEST TELEMETRY' tab, a table shows the following data:

Time	Key	Value
2019-07-28 13:22:00	Current_pv	3.779951
2019-07-28 13:22:00	Power_pv	62.36919
2019-07-28 13:22:00	Voltage_b	13.03883
2019-07-28 13:22:00	Voltage_pv	16.5

At the bottom of the table, it indicates 'Page: 1', 'Rows per page: 5', and '1 - 4 of 4'.

Prototype Design & Experimental Setup:

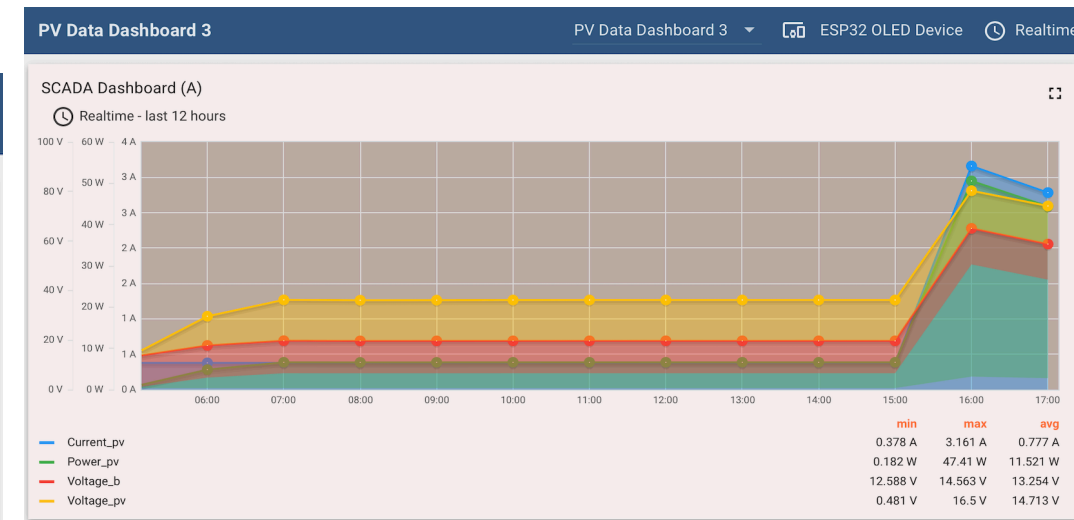
- Hardware implemented & System Set up on a Small Renewable Power Generation System (Solar PV System) at MUN ECE Lab:



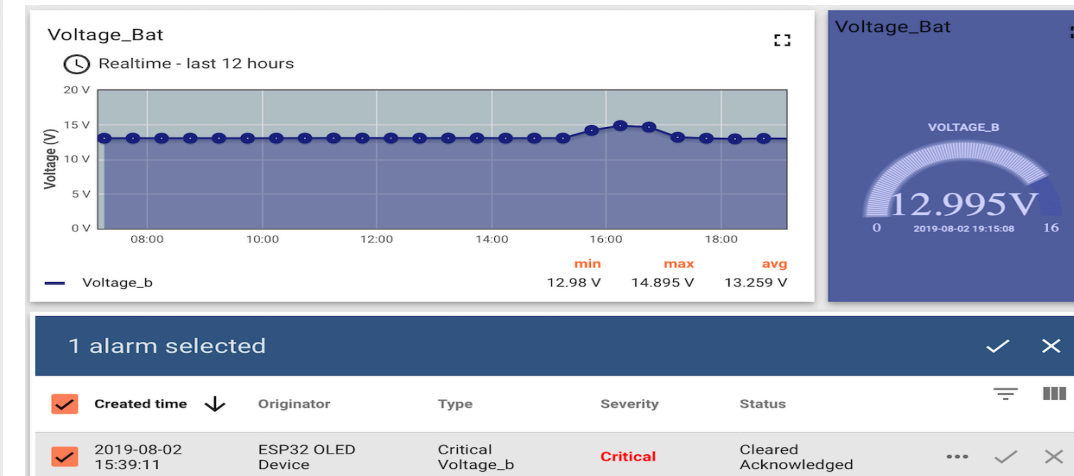
Human Machine Interface (HMI): PV Data Monitoring & Supervisory Control:

Dashboards:

Dashboards:



Alarm:



Conclusions: Some Key Features of SCADA 3:

- IoT-based, Secure, Reliable, Low-cost, Low-Power, & Open Source
- Reporting, Remote Data Monitoring & Supervisory Control, Data Acquisition, Historic Data Storage & Trending, etc.
- Local Operator Data Monitoring Interface

Table 5.1: Bill of Materials.

S/N	COMPONENT	QTY	PRICE (CAD)
1	ThingsBoard IoT Server	1	00.00
2	PostgreSQL.	1	00.00
3	Raspberry Pi 2 B	1	45.95
4	TTGO LoRa32 ESP32 OLED	1	17.49
5	16GB Memory Card	1	19.99
6	Voltage Sensor	2	11.98
7	Current Sensor	1	5.25
8	D-Link D1-524 Wireless Router	1	98.51
9	Miscellaneous (Boxes, Breadboard, Resistors, Wires, etc.)	1	80.00
Grand Total:			\$ 279.17 CAD

Table 5.2: Power Consumption of Hardware Components.

S/N	HARDWARE	POWER(W)
1	Raspberry Pi 2 B	1.8
2	ESP32 OLED (alone)	0.9
3	D-Link D1-524 Wireless Router	4.2
4	Breadboard (with ESP32 OLED, Sensors, Resistors, etc. connected)	3.3
Overall System Power Consumption (less ESP32 OLED alone):		9.3 W

Conclusions:

- Designed and Simulated a Hybrid Power System with a Small Renewable Power Generation System & ESS.
- Designed & Implemented 3 different Low-cost, Low-Power, Secure, Reliable, IoT-based Open Source SCADA Systems
- The designed Open Source SCADA systems used for a Small Renewable Power Generation System

Table 6.1: Comparison Between The Three IoT-Based Open Source SCADA Systems.

IoT-Based Open Source SCADA Systems			
Key Properties	Emoncms-based SCADA (SCADA 1)	Thinger.IO-based SCADA (SCADA 2)	ThingsBoard-based SCADA (SCADA 3)
IoT Server (MTU) RTU	Emoncms Arduino Uno/Raspberry Pi	Thinger.IO ESP32 Thing	ThingsBoard ESP32 with OLED
Communication Channel	Node-RED/Ethernet	Wi-Fi/SPI	MQTT/Wi-Fi
FIDs	Sensors	Sensors	Sensors
Security Measure	Local/Self-hosted main data server	Local/Self-hosted main data server	Local/Self-hosted main data server
Reliable from Testing?	Yes	Yes	Yes
Alarm & Notifications	Yes	Yes	Yes
Ease of Use	Yes	Yes	Yes
Internet	Yes	Yes	Yes
Offline Option	No	Yes	Yes
Local Operator Interface	No	No	Yes
Cost	\$404.84 CAD	\$291.87 CAD	\$279.17 CAD
Power Consumption	5.1 W	9.4 W	9.3 W

Key Contributions:

- A house in Nigeria chosen for the HPS Design; House Energy Assessments; Optimum Renewable Generation System Sizing.
- Implementation of SCADA System security strategies on each system.
- Selection & Testing of each SCADA Element for overall system design goals, e.g. Low-power, low-cost, reliability, etc.

List of Publications:

Refereed Journal Articles:

1. Lawrence O. Aghenta and M. Tariq Iqbal, "Design and implementation of a low- cost, open source IoT-based SCADA system using ESP32 with OLED, ThingsBoard and MQTT protocol," Submitted with *AIMS Electronics and Electrical Engineering*, October 2019 . (*Under Review*) .
2. Lawrence Oriaghe Aghenta and Mohammad Tariq Iqbal, Low-Cost, "Open Source IoT-Based SCADA System Design Using Thinger.IO and ESP32 Thing," *Electronics 2019*, 8(8), 822; <https://doi.org/10.3390/electronics8080822>.
3. Lawrence O. Aghenta and M. Tariq Iqbal, "Design and Dynamic Modelling of a Hybrid Power System for a House in Nigeria," *International Journal of Photoenergy*, Volume 2019, Article ID 6501785, 13 pages; <https://doi.org/10.1155/2019/6501785>.

Refereed Conference Publication:

1. Lawrence O. Aghenta and M. Tariq Iqbal, "Development of an IoT-Based Open Source SCADA System for PV System Monitoring," Presented at *CCECE 2019*, Edmonton, AB, Canada. May 5 - 8, 2019; doi: 10.1109/CCECE.2019.8861827

Regional Conference Publications:

1. Lawrence O. Aghenta and M. Tariq Iqbal, "A Low-Cost, Open Source IoT-Based SCADA System Design, and Implementation for Photovoltaics," Accepted for presentation at the 28th IEEE NECEC 2019, St. John's, NL, Canada. November 19, 2019.
2. Lawrence O. Aghenta and M. Tariq Iqbal, "Thermal Modelling and Analysis of a House in Nigeria and PV System Design to meet its Energy needs," Presented at the 27th IEEE NECEC 2019, St. John's, NL, Canada. November 13, 2018.

Poster Presentation:

1. Lawrence O. Aghenta and M. Tariq Iqbal, "Internet of Things (IoT) based Reliable Open Source SCADA System for Remote Battery Energy Storage Systems," Presented during the poster session at the NESTNet 2nd Annual Technical Conference, Ryerson University, Toronto, ON, Canada. June 18 - 20, 2018.

Future Recommendations:

- Detailed Reliability Calculations & Analysis of each of the Open Source SCADA Systems for Reliability Comparison.
- Inclusion of Data Encryption on each SCADA Communication Channel for greater security.
- Emoncms installed on a more recent Linux machine, while using ESP32 Thing as the RTU, and MQTT protocol for data transfer from the RTU to the MTU to improve on SCADA 1 & reduce cost.
- Configuring the Raspberry Pis in SCADA 2 & 3 as a Wireless Access Point to provide the needed Wi-Fi connections, thereby eliminating the Wi-Fi Router in each case to further reduce power consumption.
- Developing new Open Source SCADA Systems using CoAP API and HTTP API of the ThingsBoard platform for data transfer so as to compare their performance with the MQTT API used in SCADA 3 option.

Acknowledgements:

Thank you!

- My Supervisor: Professor M. Tariq Iqbal.
- Funding Support: School of Graduate Studies, Faculty of Engineering & Applied Science MUN, NSERC & NESTNet.
- My Colleagues, Friends & Family.

Acknowledgements:

Thank you!



Contributions & Questions???

Some References:

- [1]. IEC White Paper, "Electrical Energy Storage." Internet: <https://www.iec.ch/whitepaper/pdf/iecWP-energystorage-LR-en.pdf>. [Accessed on 27 August 2019].
- [2]. A. Mercurio, A. Di Giorgio and P. Cioci, "Open-Source Implementation of Monitoring and Controlling Services for EMS/SCADA Systems by Means of Web Services— IEC 61850 and IEC 61970 Standards," IEEE Transactions on Power Delivery, vol. 24, no. 3, pp. 1148-1153, July 2009. doi: 10.1109/TPWRD.2008.2008461
- [3]. K. Stouffer, J. Falco and K. Kent, "Guide to Supervisory Control and Data Acquisition (SCADA) and Industrial Control Systems Security—Recommendations of the National Institute of Standards and Technology," Special Publication 800-82, Initial Public Draft, Sept. 2006.
- [4]. White paper on SCADA Systems Overview, "Telemetry \& Remote SCADA Solutions." Available Online: www.schneider-electric.com. Document Number TBUL00001-31, March 2012. [Accessed on 2 September 2019]
- [5]. A. Sajid, H. Abbas and K. Saleem, "Cloud-Assisted IoT-Based SCADA Systems Security: A Review of the State of the Art and Future Challenges,". IEEE Access, vol. 4, pp. 1375-1384, 2016. doi: 10.1109/ACCESS.2016.2549047.
- [6]. L. Abbey, "Telemetry / SCADA Open Systems vs Proprietary Systems," Available Online: <https://www.abbey.co.nz/telemetry--scada-open-vs-proprietary-systems-2003.html> [Accessed on 4 September 2019]
- [7]. A. M. Grilo, J. Chen, M. Díaz, D. Garrido and A. Casaca, "An Integrated WSA and SCADA System for Monitoring a Critical Infrastructure," in IEEE Transactions on Industrial Informatics, vol. 10, no. 3, pp. 1755-1764, Aug. 2014. doi: 10.1109/TII.2014.2322818
- [8]. M. Zahran, Y. Atia and A. Abulmagd, "Reliable, Cheaper, and Modular New SCADA System," ResearchGate publication: <https://www.researchgate.net/publication/263374945>], July 2011.
- [9]. S. D. Antón, D. Fraunholz, C. Lipps, F. Pohl, M. Zimmermann and H. D. Schotten, "Two decades of SCADA exploitation: A brief history," 2017 IEEE Conference on Application, Information and Network Security (AINS), Miri, 2017, pp. 98-104. doi: 10.1109/AINS.2017.8270432
- [10]. S. Sahin, M. Ölmez and Y. Isler, "Microcontroller-Based Experimental Setup and Experiments for SCADA Education," IEEE Transactions on Education, vol. 53, no. 3, pp. 437-444, Aug. 2010. doi: 10.1109/TE.2009.2026739
- [11]. M. Regula, A. Otcenasova, M. Roch, R. Bodnar and M. Repak, "SCADA system with power quality monitoring in Smart Grid model," 2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC), Florence, 2016, pp. 1-5. doi: 10.1109/EEEIC.2016.7555577
- [12]. A Publication of ThingsCloud Technologies PVT Ltd, "Ultimate List of 50 IoT Platforms of 2019,". Available Online: <https://ebooks.thingsai.io/ultimate-list-of-iot-platforms>.
- [13]. G. Alamri and T. Iqbal, "Sizing of a hybrid power system for a house in Libya," 2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, 2016, pp. 1-6. doi: 10.1109/IEMCON.2016.7746365.
- [14]. T. A. Jeshaa and M. T. Iqbal, "Thermal simulation and energy consumption analysis of two houses in St. John's, Newfoundland," Procedia Engineering, 105(2015), pp. 607 – 612. <https://doi.org/10.1016/j.proeng.2015.05.038>.
- [15]. Xie Lu. "Supervisory Control and Data Acquisition System Design for CO2 Enhanced Oil Recovery", Master of Engineering Thesis, Technical Report No. UCB/EECS-2014-123. EECS Department, University of California at Berkeley, May 21, 2014.
- [16]. S.A. Boyer, "{\em SCADA: Supervisory Control and Data Acquisition}," 4th ed.; International Society of Automation: Research Triangle Park, NC, USA, 2009. Available online: [\url{https://automation.isa.org/files/chapters/SCADA-Supervisory-Control-and-Data-Acquisition-Fourth-Edition-Chapter-10.pdf}](https://automation.isa.org/files/chapters/SCADA-Supervisory-Control-and-Data-Acquisition-Fourth-Edition-Chapter-10.pdf) (accessed on 9 July 2019).
- [17]. M.S. Thomas and J. D. McDonald, "{\em Power System SCADA and Smart Grids}," CRC Press: Boca Raton, FL, USA, 19 December 2017. [On-line]. Available online: [\url{https://books.google.ca/books?id=bAhEDwAAQBAJ}](https://books.google.ca/books?id=bAhEDwAAQBAJ) (accessed on 23 June 2019).
- [18]. A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,". IEEE Communications Surveys and Tutorials, vol. 17, no. 4, pp. 2347-2376, Fourthquarter 2015. doi: 10.1109/COMST.2015.2444095.