Spring 2020

Solar Panel Monitor and Data Logger

Ryan Murray

Sharan Singh

SANTA CLARA UNIVERSITY

Department of Electrical Engineering

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

Ryan Murray, Sharan Singh

ENTITLED

SOLAR PANEL MONITOR AND DATA LOGGER

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF SCIENCE
IN
ELECTRICAL ENGINEERING

Sarah Kate Wilson
Sarah Kate Wilson (Jun 12, 2020 07:33 PDT)

Thesis Advisor date

Shoba Krishnan
Shoba Krishnan (Jun 12, 2020 07:46 PDT)

Department Chair date

SOLAR PANEL MONITOR AND DATA LOGGER

By

Ryan Murray, Sharan Singh

SENIOR DESIGN PROJECT REPORT

Submitted to
the Department of Electrical Engineering

of

SANTA CLARA UNIVERSITY

in Partial Fulfillment of the Requirements
for the degree of
Bachelor of Science in Electrical Engineering

Santa Clara, California

Spring 2020

ABSTRACT

When deploying systems halfway across the world, having some system for monitoring their performance and status is essential. All We Are, in their mission to build solar panel installations in Uganda, encounters this problem daily. For our project, we worked with All We Are to deliver a system to monitor these solar installations. Building an embedded solution that can communicate with the solar installations and upload the data to the cloud allowed us to accomplish this goal. Our system connects data from the existing solar panel infrastructure to the cloud. The platform of our system is a Raspberry Pi microprocessor. It interfaces with the existing solar infrastructure through several sensors and a serial connection to the charge controller to record data. To communicate this data to the cloud, we use another serial device, a GSM modem, to connect to a cellular network and upload to a web server.

ACKNOWLEDGMENTS

We appreciate the support from our advisor, Dr. Wilson, whose experience and guidance was essential to our work on this project. Additionally, we are grateful to Cameron Whiteman and the rest of the All We Are organization for giving us the opportunity to work on this project and contribute to their mission to change lives in Uganda.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

All We Are is a non-profit organization that aims to bring sustainable, affordable, and renewable energy solutions to communities in Uganda. Since its first installation in 2015, All We Are has completed over thirty solar installations through the help of donors and US volunteers [1]. By providing communities with sustainable power, All We Are aims to improve infrastructure and help change the lives of people in Uganda.

Because All We Are deploys installations thousands of miles away in Uganda, it is important to have a system for monitoring these installations. Currently, their solution involves hiring a local technician in Uganda. This technician drives around to each location, records data manually from the charge controller, writes down the information, and emails it to All We Are. This highly manual process wastes time and resources, and clearly leaves room for improvements in efficiency. Additionally, there is a desire to improve the coverage of the data being collected.

Currently, the only data available is data directly from the charge controller. By adding a device to interface with the inverter, we can incorporate data collection from the inverter as well. Additionally, All We Are would like the ability to temporarily disable service remotely if necessary. For example, if it is discovered that a school is breaking their contract with All We Are and new safety concerns are raised, this feature would enable them to shut off power until the issues are rectified.

With the help of a more efficient system that adds new functionality, our project team hopes to enable All We Are to have a greater impact with their resources.

II. OBJECTIVES

The primary objective of this project is to provide a system to enable remote monitoring of the solar installations All We Are maintains in Uganda. To achieve this goal, we need to (i) establish a proof-of-concept of an embedded system infrastructure upon which we can build out the project, (ii) develop an interface to the solar installation via sensors and communication with the charge controller, (iii) provide a solution for connecting to the Internet so that data can be uploaded, and (iv) a cloud database to store and present data collected by the system.

Figure 1: The TS-M-2 charge controller module, the primary source of data to the system [2].

III. ADDITIONAL PROJECT REQUIREMENTS

Because All We Are is a non-profit, cost is a major concern for this project. Due to the need to maintain a consistent cellular connection, there are added upkeep costs in addition to the initial equipment costs. As such, one of the primary requirements for this project is to minimize both the up-front costs and the ongoing upkeep costs to make this system as financially sustainable and efficient as possible.

Additionally, one of All We Are's guiding principles is that hardware components are sourced locally to ensure that the money is going back into the local communities that they are trying to help. To comply with this, we want to ensure that wherever possible, we are designing our system with components that can be sourced locally in Uganda.

Another guiding constraint is the ability to support modular systems. Because All We Are works with a variety of schools with a variety of solar needs, there is diversity in the structure of some of their solar installations. To accommodate this, our system needs to be able to work with different setups at the schools in Uganda. For example, our system may need to support a school with multiple charge controllers in multiple locations. Finally, because the scope of this project is limited to building a proof-of-concept system, All We Are will need to transition the system to a more finalized, production-ready system at a later date. To ensure that this process goes as smoothly as possible, we need to take extra care to emphasize good design principles and documentation from the ground up to make as maintainable a system as possible.

IV. PROJECT PLAN

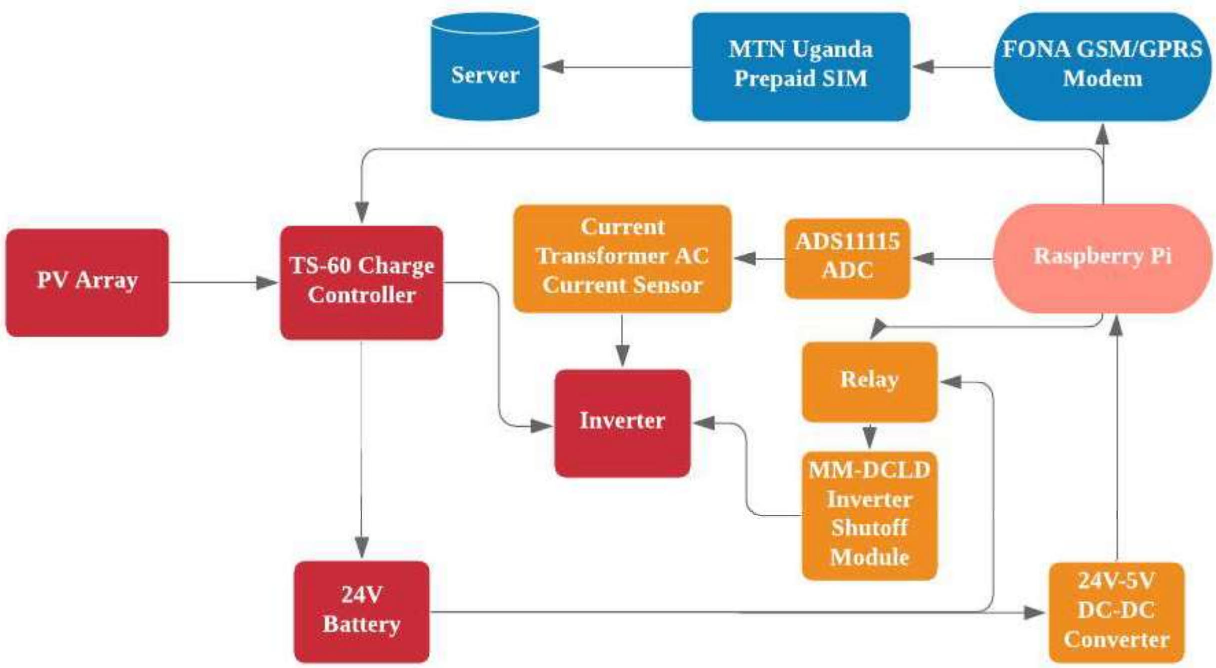


Figure 2: The high-level block diagram for our system. Colored in red is the existing solar infrastructure, in orange is the hardware to interface with the existing system, and in blue are the components of the networking system.

For the basis of our embedded system, we elected to use a Raspberry Pi. We chose the Raspberry Pi over comparable boards for a few reasons. First, it is widely available, making eventual sourcing in Uganda a much more realistic possibility than with some of the alternatives. Second, its immense popularity means that the learning curve will be minimal when handing the project off to All We Are to further develop. The popularity of the Raspberry Pi also means that it is widely supported and will have minimal compatibility issues with other potential hardware components. Finally, the Raspberry Pi runs a Linux operating system. This is extremely powerful and offers an advantage over more lightweight microcontrollers such as the Arduino.

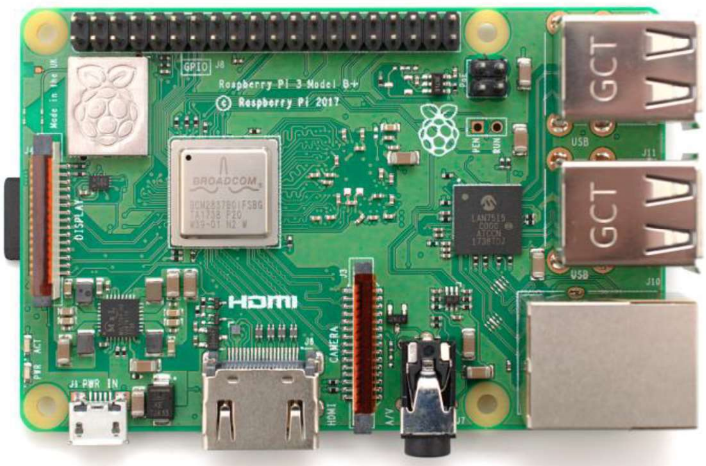


Figure 3: A Raspberry Pi, the computer serving as the basis for our system [3].

One major decision we had to make for our project plan was choosing a method of connecting to the Internet. To accomplish this, we decided to use a serial modem that would connect to a GSM [4] cellular network. We settled on a GSM modem by comparing cellular coverage in Uganda with the locations of the schools All We Are supports, as can be seen below. We concluded that a GSM/GPRS modem would have better coverage of the more rural school locations than 3G or LTE while still meeting our minimal needs for data transfer rate. To actually connect to the Internet from the modem, we used PPPD, Linux's Point-to-Point Protocol daemon, to establish a Point-to-Point Protocol (PPP) connection. After configuring this tool, we were able to use the modem as the primary network connection on the Raspberry Pi.

Figure 4: A map of the schools All We Are supports. Many are located in and around Kampala [1].

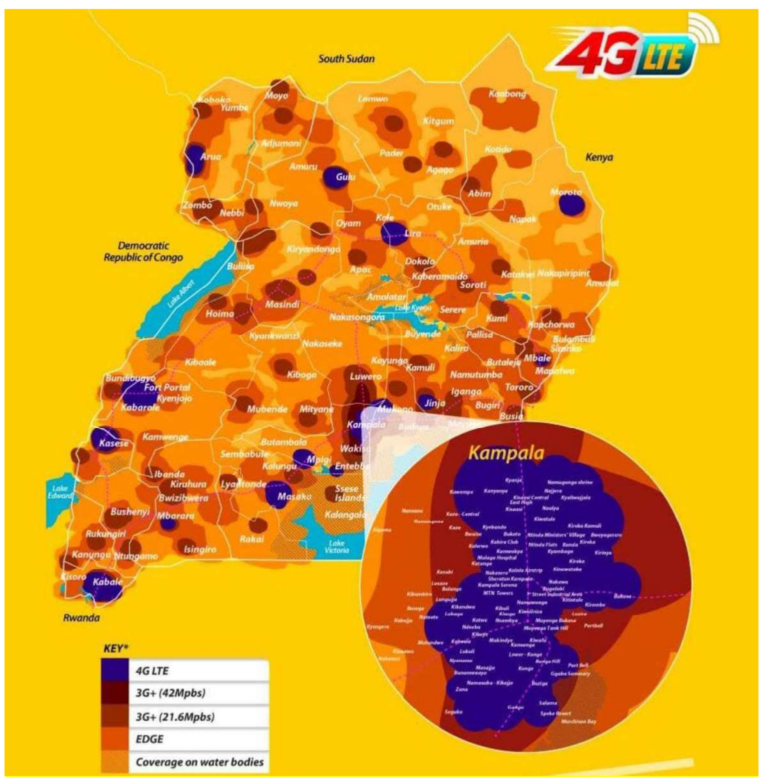


Figure 5: A cellular coverage map from MTN Uganda, one of the major cell carriers in Uganda. Note the lack of 3G or LTE coverage in the more rural areas [5].

Another major component of this system is the interface to the TS-60 solar charge controller. This interface is the primary source of data collection for our system. We made use of the Raspberry Pi's many available serial interfaces [6] to establish a Modbus connection to the charge controller. We then used PyModbus, a popular Python library to manage Modbus connections, to issue commands to the charge controller. Through PyModbus, we were able to read from designated memory locations on the charge controller, outlined below, to pull the data we needed. In doing so, we were able to completely monitor the charge controller, tracking battery voltage, various temperatures, solar panel output, and much more.

RAM

PDU Addr	Logical Addr	Variable name	Variable description	Units	Scaling or Range
0x0000	1		Internal Use		
0x0001	2		Internal Use		
0x0002	3		Internal Use		
0x0003	4		Internal Use		
0x0004	5		Internal Use		
0x0005	6		Internal Use		
0x0006	7		Internal Use		
0x0007	8		Internal Use		
0x0008	9	adc_vb_f	Battery voltage, filtered ($\tau \approx 2.5$s)	V	$n \cdot 96.667 \cdot 2^{-15}$
0x0009	10	adc_vs_f	Battery sense voltage, filtered ($\tau \approx 2.5$s)	V	$n \cdot 96.667 \cdot 2^{-15}$
0x000A	11	adc_vx_f	Array/Load voltage, filtered ($\tau \approx 2.5$s)	V	$n \cdot 139.15 \cdot 2^{-15}$
0x000B	12	adc_ipv_f	Charging current, filtered ($\tau \approx 2.5$s)	A	$n \cdot 66.667 \cdot 2^{-15}$
0x000C	13	adc_iload_f	Load current, filtered ($\tau \approx 2.5$s)	A	$n \cdot 316.67 \cdot 2^{-15}$
0x000D	14	Vb_f	Battery voltage, slow filter ($\tau \approx 25$s)	V	$n \cdot 96.667 \cdot 2^{-15}$
0x000E	15	T_hs	Heatsink temperature	°C	-128 to +127
0x000F	16	T_batt	Battery temperature (RTS connected) (0x80 if not connected)	°C	-128 to +127
0x0010	17	V_ref	Charge regulator reference voltage	V	$n \cdot 96.667 \cdot 2^{-15}$
0x0011	18	Ah_r_HI	Ah resetable, HI word	Ah	$n \cdot 0.1$
0x0012	19	Ah_r_LO	Ah resetable, LO word	-	
0x0013	20	Ah_t_HI	Ah total, HI word	Ah	$n \cdot 0.1$
0x0014	21	Ah_t_LO	Ah total, LO word	-	
0x0015	22	hourmeter_HI	hourmeter, HI word	h	0 to $(2^{24}-1)$

Figure 6: A list of the memory addresses used to pull data from the charge controller [7].

To expand upon the data currently being collected by All We Are's current manual system, we wanted to add monitoring to the inverter output. We achieved this by adding a current transformer current sensor [8] that can be installed at the inverter output. This current transformer is clamped around one of the inverter leads so that the current is passing through the transformer. The transformer then produces an AC current proportional to the current passing through it. We are then able to measure this current and calculate the inverter current from the sense resistor value and the transformer's turn ratio. To measure this signal, we applied a filter capacitor and a DC bias with a voltage divider, as shown below, then sampled the voltage across a sense resistor using an I2C ADC. The value for the sense resistor to maximize ADC resolution while keeping the voltage between GND and VDD can be found with the following calculation:

$$R_{sense} = \frac{VDD * N}{2 * I_{peak}}$$

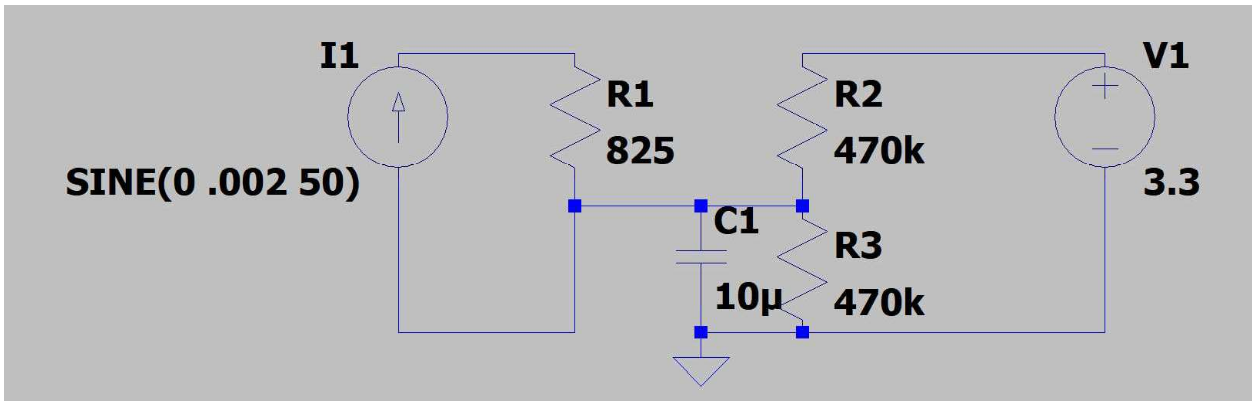


Figure 7: The circuit to prepare the transformer output for the ADC. The R1 sense resistor value is chosen to maximize ADC resolution at a peak primary current of 4 A.

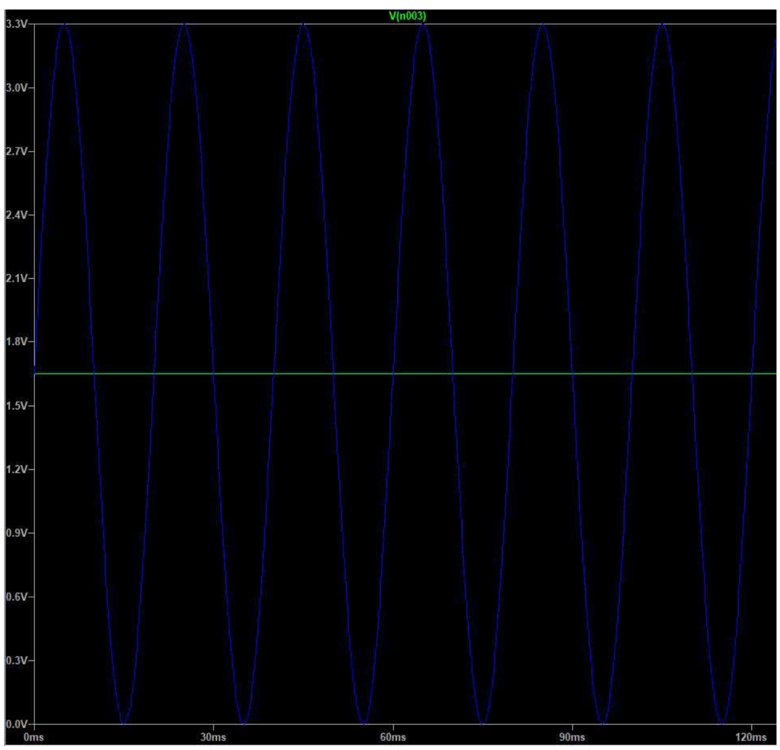
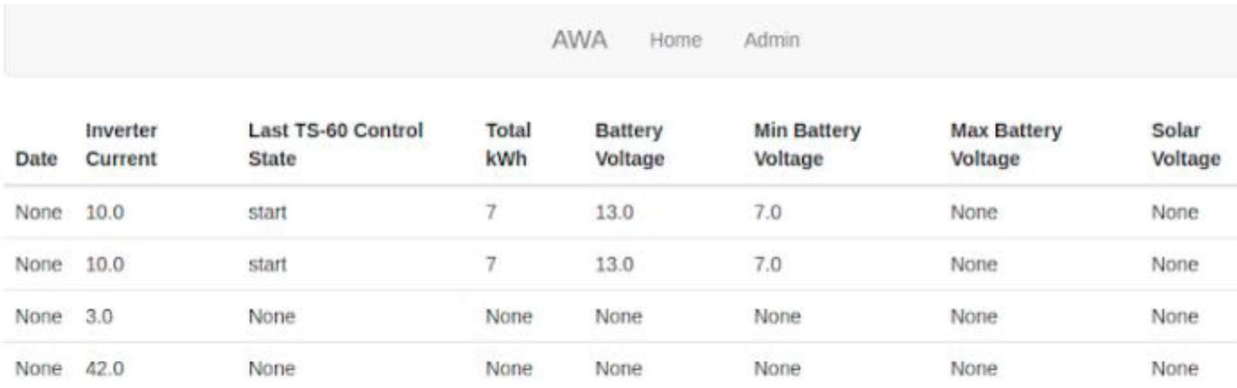


Figure 8: The simulated voltage across the sense resistor. The voltage oscillates between GND and VDD.

Another major improvement our project adds to All We Are's system is the ability to remotely disable the inverter. The Magnum MM1324E, the inverter model used at all of All We Are's schools, is configured by the manufacturer to shut off when a 24 V signal is removed. To control this from the Raspberry Pi, we used a relay module. By adding a relay, the Raspberry Pi is able to use a low voltage, low current signal from the on-board GPIO pins to switch on and off the signal to the inverter. Additionally, the relay switch is set to normally-closed (NC) to ensure that the inverter is not mistakenly shut off if the Raspberry Pi loses power.

To actually upload the collected data and receive shutoff commands from the Internet, we developed two functional solutions. For early stage testing and validation, we uploaded data to a spreadsheet through the Google Sheets API.. This initial solution was later replaced by a custom web server that collects data through the Raspberry Pi. As an early stage prototype, the Google Sheets solution enabled us to easily transfer information to and from the Raspberry Pi and was our primary upload solution for our testing in Uganda since the web server had not yet been completed.

To build the web server, we used Flask, a lightweight Python web framework. Flask has a healthy ecosystem of extensions, which enabled us to quickly bring up a functional website. For example, the Flask-SQLAlchemy extension allowed our web application to use the SQLAlchemy object-relational mapper (ORM) [9] to easily build and interact with a database. In this database, we created tables to track all registered installations, the data associated with those installations, and the users authorized to read data and send commands to the installations. We also used several other Flask extensions: Flask-Login to manage user sessions and Flask-Bootstrap to incorporate Bootstrap CSS templates into our website. After creating the basic interface and functionality we needed to implement a secure API so that the Raspberry Pi could interact with the server. To achieve this, each installation when registered is issued an API key and an API secret. This key-secret pair can be revoked at any time and gives the user permission to upload data only for that specific installation. These credentials are then used to request a JSON Web Token (JWT) [10] from the server. JWT offers a few major advantages for our application. First, passing a token to the server with every request instead of the API secret as would happen with HTTP basic authentication minimizes the number of opportunities for the API secret to be compromised. Additionally, a JWT contains a signature created from the header and payload, which ensures that the JWT has not been tampered with. Using JWT allows us to verify that (i) each API request is coming from an authorized user and (ii) the user is authorized to access the resource they are attempting to access. After receiving a JWT from the server, the client simply attaches the token to all future API requests until the token expires.



Date	Inverter Current	Last TS-60 Control State	Total kWh	Battery Voltage	Min Battery Voltage	Max Battery Voltage	Solar Voltage
None	10.0	start	7	13.0	7.0	None	None
None	10.0	start	7	13.0	7.0	None	None
None	3.0	None	None	None	None	None	None
None	42.0	None	None	None	None	None	None

Figure 9: The basic UI of the website for presenting installation data, filled with sample data for testing purposes.

As for the hosting implementation of this web server, the site was hosted on both Amazon Web Services (AWS) as well as the Google Cloud Platform (GCP) for testing purposes. For both the GCP and AWS setup, the web application directory along with an *app.yaml* file and a *requirements.txt* file were uploaded from a personal computer. The *app.yaml* file configured the settings of the Cloud App Engine and the *requirements.txt* file included a list of all the libraries required for the web application to function. The Gunicorn framework was then used to run the Flask application on the server. After following these simple steps, our website was now easily accessible from a public domain.

V. PERFORMANCE AND RELIABILITY

Every year, All We Are travels to Uganda in early March so several members of the organization can conduct site visits and help plan the organization's next steps. This presented an opportunity to perform some early testing on-site at an actual school in Uganda. For this reason we had to have a completed fully functional prototype ready to hand off to All We Are in mid February.

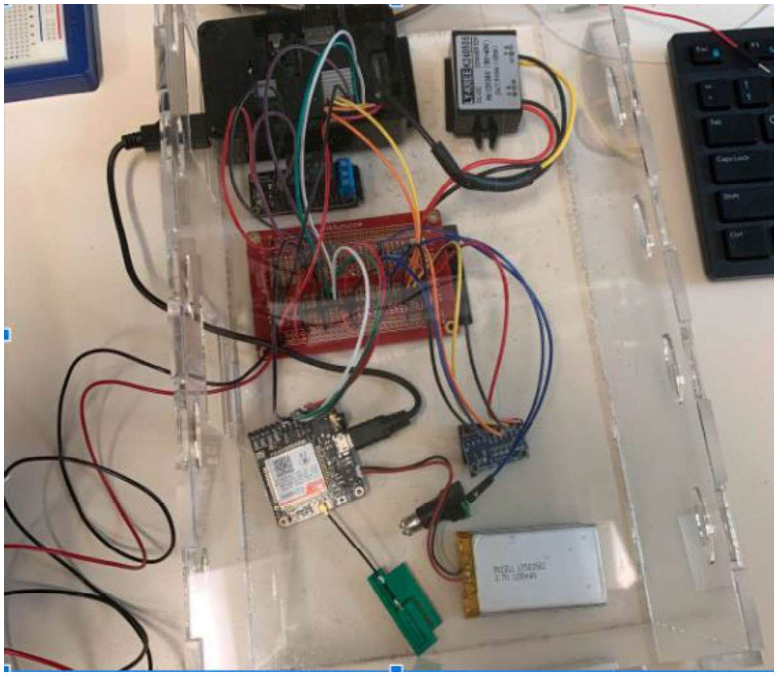


Figure 10: The completed prototype that was sent to Uganda. From top to bottom is the Raspberry Pi and power supply, relay module, circuit board, and the modem and ADC.

Our testing at Santa Clara prior to delivering the prototype to Uganda was highly successful. We thoroughly tested each subsystem, with extra attention paid to the network connection and charge controller interface, two of the core components of our project. The connection to the cellular network was extremely consistent and more than sufficient for our upload needs. Similarly, the charge controller interface was extremely effective and was able to consistently read data from the charge controller. After testing each individual subsystem, we then conducted integration testing. We ran the complete system for an extended period of time to ensure that the system would be able to function in the field. Despite the limited amount of testing time we had available due to the early time frame of the Uganda trip, we were able to validate both the individual components of the system as well as their interoperability.

While our testing at SCU was very successful, our testing on the ground in Uganda produced mixed results. Despite the consistent performance of the GSM modem here in the US, we encountered problems with this project subsystem when testing live at a school in Uganda. Our biggest issue revolved around the modem being unable to reliably maintain a connection to the Internet over the cellular network. This would result in frequent latency spikes, sometimes taking tens of seconds to receive a reply from the server. Such latency spikes rendered our cellular connection unusable for a consistent upload as the requests would time out. Due to the short time window for testing our project at the school, we were unable to fully resolve the issue through remote debugging. Since that time we have developed some potential solutions that will be discussed in the "Future Work" section of this thesis. Fortunately, despite the issues that arose with the cellular connection, the other subsystems of our project were successfully tested. Both the charge controller interface and the current transformer sensor for measuring the inverter output, two crucial sources of data collection for our system, were able to consistently retrieve data from the solar installation at the school. This successfully tested functionality ensured that there is a clear path towards a polished final product after improving upon the GSM modem functionality and establishing a more reliable Internet connection.

In terms of the reliability of our system, the modem issues clearly raise some concerns. Without a consistent and reliable Internet connection, the entire system cannot reliably function. However, we are confident that with more time spent developing the networking solution, we can increase its reliability to make the system overall more reliable. Regarding the rest of the system, our proof-of-concept performed well in terms of reliability. Some minor issues arose with the serial connection to the modem failing, likely caused by disturbances during transportation. Again, we are confident that this can be resolved to minimize reliability concerns and prevent any issues during operation. Overall, after further development our proof-of-concept should easily be able to be expanded into a reliable final product.

VI. PROFESSIONAL ISSUES AND CONSTRAINTS

In reviewing our project, we have noted some potential economic, health and safety, and usability considerations. As previously stated, one of the organization's goals is to use components purchased locally in an effort to to support the economy. However, we struggled to fully source all of our hardware from Uganda because some of the specialized materials from the project are only available through certain sellers in the US. For example, the Raspberry Pi is something we will be able to source from Uganda, but some of the more specialized hardware, such as the specific model of GSM modem may be difficult to source locally. As such, part of the future work of this project will be to revise the bill of materials where possible to better fit the sourcing preferences of All We Are. In regards to health and safety, one potential hazard is the voltages present if working with the inverter when installing the system. To minimize the risk associated with this hazard, the only people handling the inverter will be professionals trained to work with high voltages. Additionally, the current transformer we chose to measure the inverter output is non-invasive, meaning that it can be installed safely without directly contacting the inverter leads. Finally, usability is one of the areas of improvement as identified after our initial test in Uganda in early March. In trying to implement the system, our contact at All We Are reported that some parts of the process to set up and use the system were unclear. After returning, our team has begun improving the documentation process for our project to ensure that there is a smooth transition when handing off the project to All We Are.

In addition to these concerns, we also have to consider the societal impact of our work and the civic engagement it entails. Due to the noble goals of All We Are, our project has the potential to have a great impact on the communities it serves. A successful project will mean that All We Are is able to expand further, bringing sustainable power to more schools in Uganda. However, this increases the risk for unintended consequences if the project is flawed. For example, delivering an unreliable system could set the organization back and hurt their mission to do good in Uganda. Regarding civic engagement, there are minimal considerations for this project. The only major area of concern involves local electrical codes. In modifying the All We Are system, we need to ensure that our additions comply with local regulations.

VII. CONCLUSION AND FUTURE WORK

Overall, the project successfully met our goal of creating a proof-of-concept system to remotely monitor All We Are's solar installations. We were able to create a fully functional prototype by mid-February that was sent to Uganda with All We Are to conduct testing on-site at a school. The testing produced mixed results, but there is a clear path forward to improving the system into a final product that can meet the needs of All We Are at all of their sites. In addition to developing the hardware solution to accomplish this, we also built out a custom website that can retrieve the data being uploaded from each installation through a secure API and display it.

One of the major remaining items to work on in the future is resolving the networking issues discovered while testing in Uganda. Our group has outlined a few potential solutions and causes to investigate. For one, we would like to investigate our modem's sensitivity to fade spots, areas of relatively weak signal strength. We would also like to research alternative antennas that may provide our system with a better network connection. Another possible solution involves switching to a different cell carrier that might have better signal strength at the school locations. Finally, we can consider switching to a 3G modem for better speeds in some areas, sacrificing coverage in rural areas for a better connection in urban ones.

In addition to resolving existing issues, there will be an ongoing effort within All We Are to transition our project to a more production-ready final product. Part of this effort involves working with another senior design team from NC State University, who worked on physical device security at All We Are sites, to integrate our projects. To accomplish this, our group will be volunteering with All We Are in some capacity to see that the transition goes smoothly and offer any assistance we can.

VIII. REFERENCES

[1] Solarize Uganda Now. (n.d.). *All We Are*. https://allweare.org/solar/.
[2] Morningstar TS-M-2 TriStar Digital Meter LCD RJ-11 Connectors. (n.d.). *Amazon*. https://www.amazon.com/Morningstar-TS-M-2-TriStar-Digital-Connectors/dp/B007NMWVVE.
[3] Raspberry Pi 3 B+. (n.d.). *Wikimedia.* https://commons.wikimedia.org/wiki/File:Raspberry_Pi_3_B%2B_(39906369025).png.
[4] C. Hamling, "An overview of GSM data services," *IEE Colloquium on Roaming with Data*, London, UK, 1995, pp. 2/1-2/8, doi: 10.1049/ic:19950305.
[5] Coverage Map. (n.d.). *MTN Uganda*. https://www.mtn.co.ug/insight/coverage-map/.
[6] A. D. Deshmukh and U. B. Shinde, "A low cost environment monitoring system using raspberry Pi and arduino with Zigbee," *2016 International Conference on Inventive Computation Technologies (ICICT)*, Coimbatore, 2016, pp. 1-6, doi: 10.1109/INVENTIVE.2016.7830096.
[7] TriStar MODBUS Specification. (2012, September 6). *Stellavolta*. https://www.stellavolta.com/content/MSCTSModbusCommunication.pdf.
[8] K. Zuo, G. Geng and Q. Jiang, "The Application of Non-Invasive Sensor in Unbalanced Three-Phase Current Detecting," *2019 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia)*, Chengdu, China, 2019, pp. 1221-1225, doi: 10.1109/ISGT-Asia.2019.8881558.
[9] J. Armas, P. Navas, T. Mayorga, P. Rengifo and B. Arévalo, "Optimization of code lines and time of access to information through object-relational mapping (ORM) using alternative tools of connection to database management systems (DBMS)," *2017 2nd International Conference on System Reliability and Safety (ICSRS)*, Milan, 2017, pp. 500-504, doi: 10.1109/ICSRS.2017.8272872.
[10] JSON Web Tokens Introduction. (n.d.). *Auth0. https://jwt.io/introduction/*.

IX. APPENDIX A: SENIOR DESIGN CONFERENCE SLIDES

SANTA CLARA UNIVERSITY
School of Engineering

Solar Panel Monitor and Data Logger

Ryan Murray and Sharan Singh

Advisor: Dr. Wilson

Santa Clara University 1

SANTA CLARA UNIVERSITY
School of Engineering

Outline

- Background
- Objectives
- Project Plan
- Project Outcomes
- Performance and Reliability
- Project Schedule
- Summary

Santa Clara University 2

Background

- "All We Are" is a non-profit installing solar panels in Uganda
- Want to efficiently monitor data from sites
- Looking to reduce costs and effort

ALL WE ARE

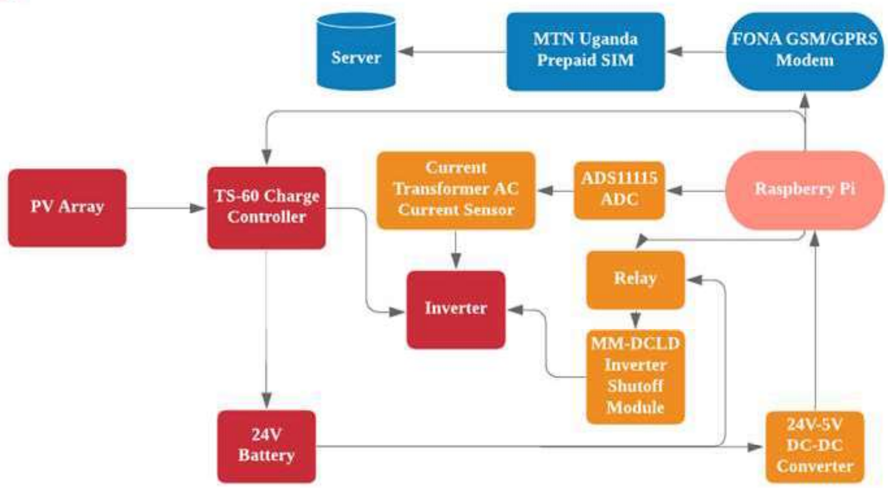
SANTA CLARA UNIVERSITY
School of Engineering

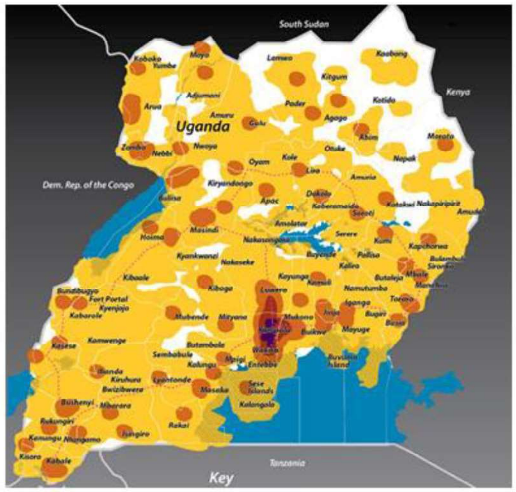
Objectives

- Monitoring system to record and upload data
- Support deployment/testing in Uganda
- Budget solution with low upkeep
- Provide documentation and support to continue development

Project Plan

- Embedded system to interface with existing installation
  - Log data
  - Upload capability
- Raspberry Pi
  - GSM modem
    - Prepaid cell plan
  - Python
- Host web server on AWS/Azure/GCP





Outline

- Background
- Objectives
- Project Plan
- Project Outcomes
- Performance and Reliability
- Project Schedule
- Summary

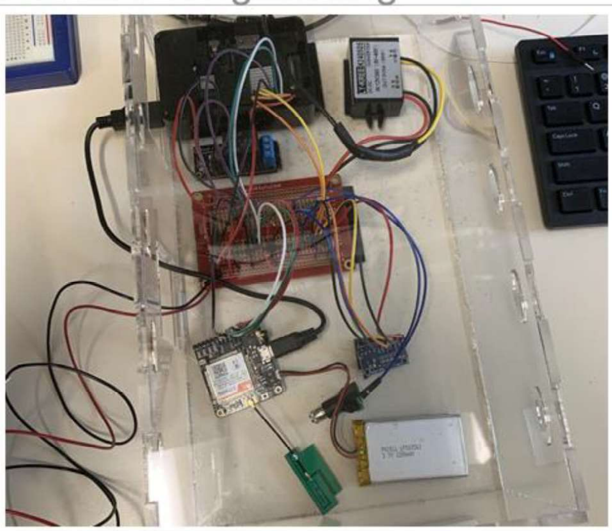
SANTA CLARA UNIVERSITY
School of Engineering

Project Outcomes

- Developed a fully compatible embedded device that interfaces with existing system and uploads all the data received onto an online database
- Supports remote shut off via request from server
- Allow for real time view of all data from each location from the solar panels
- Web server built out to receive, store and display data

Santa Clara University 9

SANTA CLARA UNIVERSITY
School of Engineering



Santa Clara University 10

Hardware Outcomes

- System is able to interface with the charge controller and inverter and reliably communicate and collect data
- Uses serial GSM modem to transmit and receive data from cellular network
  - Establishes Point to Point Protocol (PPP) to connect to Internet over this cellular connection
- Contains relay to shut off the inverter

SANTA CLARA UNIVERSITY
School of Engineering

Web Server

- Built web server using Flask, a lightweight Python web framework
- Stores and displays data from each installation
- API secured with JSON Web Tokens to upload data from installations
- To be hosted on AWS/Azure/GCP

Flask
web development,
one drop at a time

X. APPENDIX B: INFO ON EQUIPMENT AND SOFTWARE USED

Raspberry Pi: https://www.raspberrypi.org/
FONA GSM modem: https://learn.adafruit.com/adafruit-fona-mini-gsm-gprs-cellular-phone-module
TS-60 charge controller: https://www.morningstarcorp.com/products/tristar/
ADS1115 ADC: http://www.ti.com/lit/ds/symlink/ads1114.pdf?ts=1591395948761
Python: https://www.python.org/
PyModbus: https://pymodbus.readthedocs.io/en/latest/
RPi.GPIO: https://pypi.org/project/RPi.GPIO/
Smbus2: https://pypi.org/project/smbus2/
Flask: https://flask.palletsprojects.com/en/1.1.x/
Flask-Login: https://flask-login.readthedocs.io/en/latest/
Flask-SQLAlchemy: https://flask-sqlalchemy.palletsprojects.com/en/2.x/
Flask-JWT: https://pythonhosted.org/Flask-JWT/
Flask-Bootstrap: https://pythonhosted.org/Flask-Bootstrap/