US Army Tactical Microgrid System Civilian Energy Storage System

By

Sahil Morrow

Matthew Weberski

Patrick Yang

Final Report for ECE 445, Senior Design, Spring 2020

TA: Vassily Petrov

8 May 2020

Project No. 49

Abstract

The United States Army Corps of Engineers developed the Tactical Microgrid System for use in military bases. The Tactical Microgrid System uses diesel generators to deliver power. This design is not suitable for civilian applications because the use of diesel generators is not a sustainable long term solution. Our solution utilizes solar energy as the main source of power for the microgrid, using the diesel generators as backup. In order for this design to work, an energy storage system needs to be in place to provide energy for short periods of time. The original Tactical Microgrid System did not have a design for a battery bank or a battery management system. Adding our energy storage system to the current design of the Tactical Microgrid System would make the microgrid a long term solution for areas that have poor power plant infrastructures, or no infrastructure in place at all.

Contents

1 Second Project Motivation

1.1 Problem Statement

The Tactical Microgrid System is a decentralized power distribution system for deployment in areas that need additional means of producing electricity. There are many places around the world that lack the ability to effectively provide power to its citizens, due to limited resources or a natural disaster. The initial design of the civilian version of the Tactical Microgrid System will be deployed Guayabota, Puerto Rico. The city of Guayabota has access to the main power grid in Puerto Rico, but the issue is that it is currently operating near max capacity due to recent Earthquakes. The citizens of Guayabota get their water from the city's water pumping system, so we will aim to have our microgrid design power the water pump to lessen the strain of the main power grid and ensure people will continue to have a reliable water supply.

There is a solar farm in Guayabota that will provide us with a sustainable and long term source of energy. The solar panels only provide power intermittently, and that is why battery storage is required to decrease the amount of time the diesel generators would be on. Guayabota does not have an energy storage system available, which is why we must design the battery bank to be sent with the microgrid. Our design must include a battery bank and a communication module to serve as the microgrid battery management system. Integrating our design with the Tactical Microgrid System and the Guayabota solar farm will provide the city a reliable power distribution system.

1.2 Solution

The energy storage system design can be broken down into two components. The first component is the battery bank, and the second is the communication module. Starting with the battery bank design. It will consist of 46 Lithium Ion batteries, which will provide enough energy to power the water pumping system for approximately 12 hours. This battery bank, in combination with the Guayabota solar farm, the use of the diesel generators would be kept for emergency situations. The battery bank will have to be designed in a housing that maintains a safe operating temperature for the Lithium Ion batteries under any load conditions, and also while charging.

The second component of our energy storage system is the communication module. The Tactical Microgrid System is programmed using the TMS-DDS communication protocol. The DDS protocol is how the main microgrid controller directs power from the input sources to the load connected to the microgrid. Incorporating solar panels and battery storage means that the microgrid controller must know the charge state of the battery bank so that power through the distribution boxes can be directed to either charge the batteries, or use the energy available from the batteries. Our communication module will monitor the state of the battery bank, and send this information via UDP data packets that can follow the TMS-DDS protocol and can be interpreted by the microgrid controller.

As mentioned previously, the Tactical Microgrid System does not have the ability to incorporate battery storage or renewable energy sources. There are solutions that exist for systems that use solar energy and come complete with battery backup and battery management systems. One example is from a company Iron Edison, and their solution that offers 4.3 kWh of energy costs $10,407.30 [8]. This specific solution does not offer the amount of energy that the Guayabota water pumping system requires. Another advantage for our design is the fact that it will be immediately compatible with the TMS-DDS protocol. Taking advantage of the Tactical Microgrid System will help keep our solution cost effective, and the power distribution decisions can be kept centralized to the main microgrid controller.

1.3 High-Level Requirements

- The energy storage system is able to monitor and send its status to the Microgrid Controller through an ethernet connection through UDP
- The energy storage system supports all published and subscribed topics specified by the TMS-DDS STOR protocol
- The energy storage system has a physical user interface that toggles the energy output (on/off) and displays its current status

1.4 Visual Aid

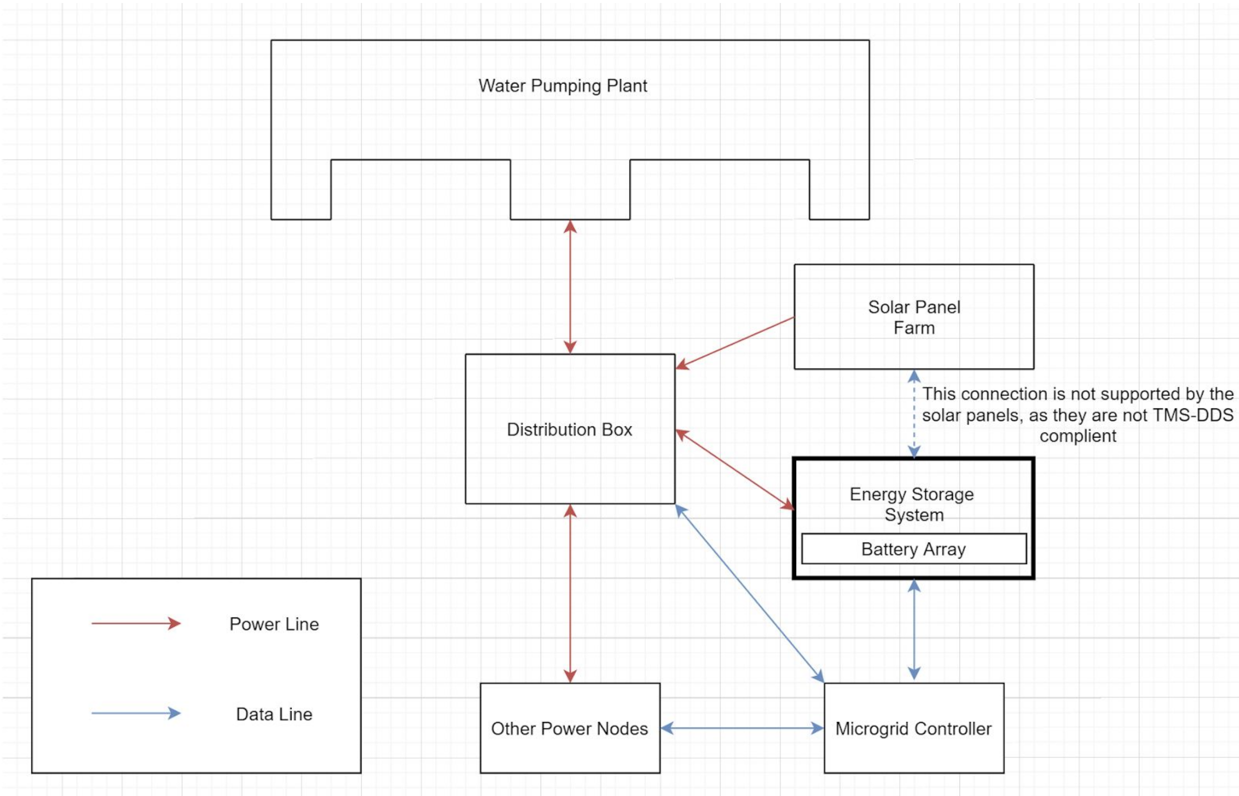


Figure 1: Overall layout of the microgrid system

1.5 Block Diagram

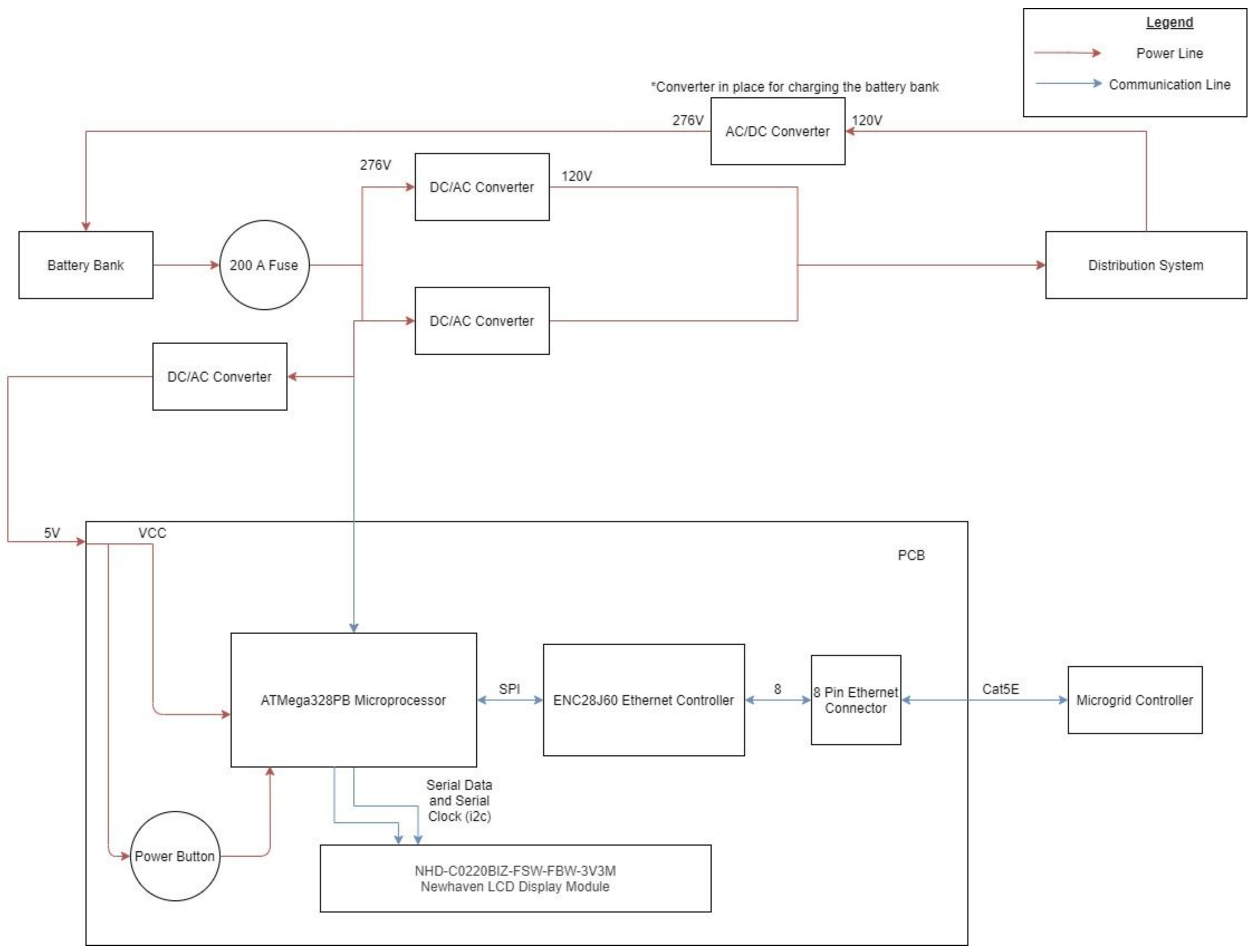


Figure 2: Functional Block Diagram

2 Second Project Implementation

2.1 Implementation Details and Analysis

2.1.1 Battery Bank

In order for solar energy to be effectively utilized in the Tactical Microgrid System design, a battery storage system is needed to provide energy during periods of time where the solar panels are not able to generate any energy. For example, solar panels will not be able to provide any energy for the water pumping station overnight when there is no sun. The first step to sizing the battery bank is to estimate the energy requirement of the water pumping station in Guayabota. We had to make multiple assumptions to get an initial estimate of water pumps energy consumption. First, the National Environmental Education Foundation estimates that Puerto Rico residents require 62 gallons of water per day [6]. Next, one acre-foot of water is equivalent to 325,851 gallons of water. Finally, the energy consumption to pump one acre-foot of water one foot is 1.71 kWh, operating at 60% efficiency [7]. Combining information with the depth of groundwater and the population of Guayabota, the energy consumption result is shown in Equation 1.

$$\frac{62\ gal/person{-}day * 3000\ people}{325851\ gal} * 1.71kWh * 56.5ft = 55.15\ kWh/day$$

Equation 1: Equation to solve estimated energy consumption of a water pumping plant.

The next step of the battery bank model is the assumptions of what type of battery we will use, and the amount of time the batteries will provide energy to the water pump. After discussion between our team and the ABE team, we are deciding to provide enough battery storage to power the water pump at the maximum rate of energy consumption for 12 hours. The reason we chose 12 hours is that we wanted to backup the solar panels overnight and during short periods of time where the sun is blocked. We made the assumption that any period over 12 hours would warrant the use of the diesel generators to turn on to power the water pump. This will allow us to balance the cost of the battery bank and take advantage of the diesel generator's availability.

For the choice of battery type, our options are either Lithium Ion or Lead Acid batteries. We chose to go with the Lithium Ion batteries for our design. Lithium Ion batteries have the advantage of a longer lifespan, especially in hot climates. Figure 3 shows a plot that compares the life cycle of Lithium Ion and Lead Acid batteries at different discharge percentages [9]. The hotter climate does present the threat that the Lithium Ion batteries would reach critical operating temperatures. To mitigate the chances of this occurring, we will implement cooling systems inside the battery bank storage case, and add temperature sensors that will closely monitor the batteries. Finally, resupplying batteries to Guayabota would be difficult and costly, since it is in a remote location of Puerto RicoWhile Lithium Ion batteries have a large upfront cost, we believe that this cost will be made up in the long term due to the longer life cycle.

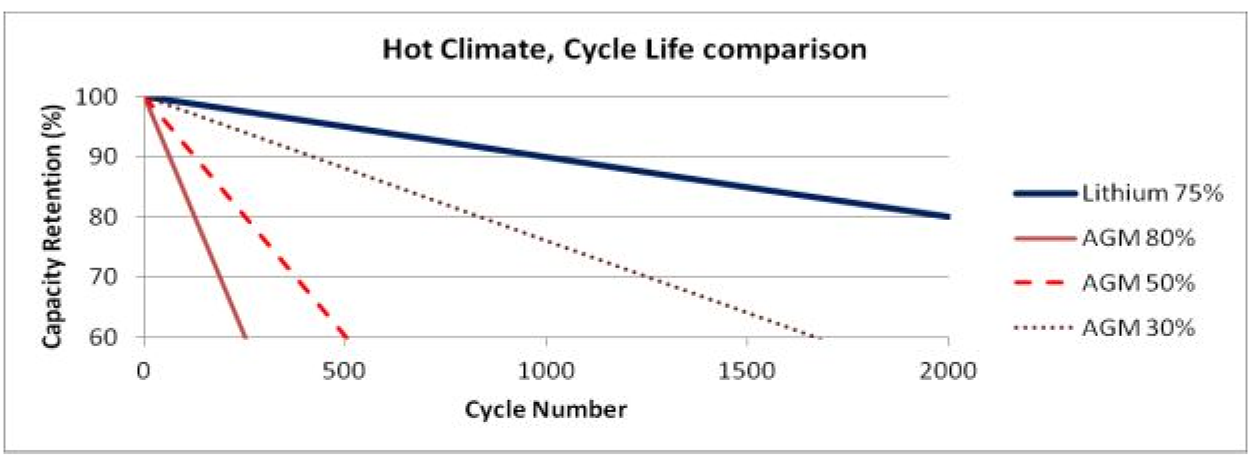


Figure 3: Battery Lifetime Comparison[9]

We can calculate the number of Lithium Ion batteries required for the Tactical Microgrid System using Equation 1. To solve for the number of batteries, we need to take the battery voltage and multiply it by the discharge rate with a depth of 50%. Because our calculation uses 50% depth of discharge, a factor of two must be added to ensure that the charge of the batteries would not be depleted prior to the 12 hour timespan.

$$\frac{2*kWh*\#\ of\ days}{Battery\ Voltage*Battery\ Discharge\ Rate}*1000 = \#\ of\ Batteries$$

$$\frac{2*55.15kWh*0.5\ days}{12V*100AH}*1000 = 46\ Batteries$$

Equation 1: Calculation for the size of the battery bank.

Our battery bank will require a large number of batteries, so it will be essential to have a storage container capable of housing every battery. Care must be taken to ensure that enclosing the batteries will not cause dramatic increases in temperature during charging and discharging, which is why we have added multiple features for air ventilation and cooling. Figure 4 shows an annotated image of the physical design of the battery bank storage case.

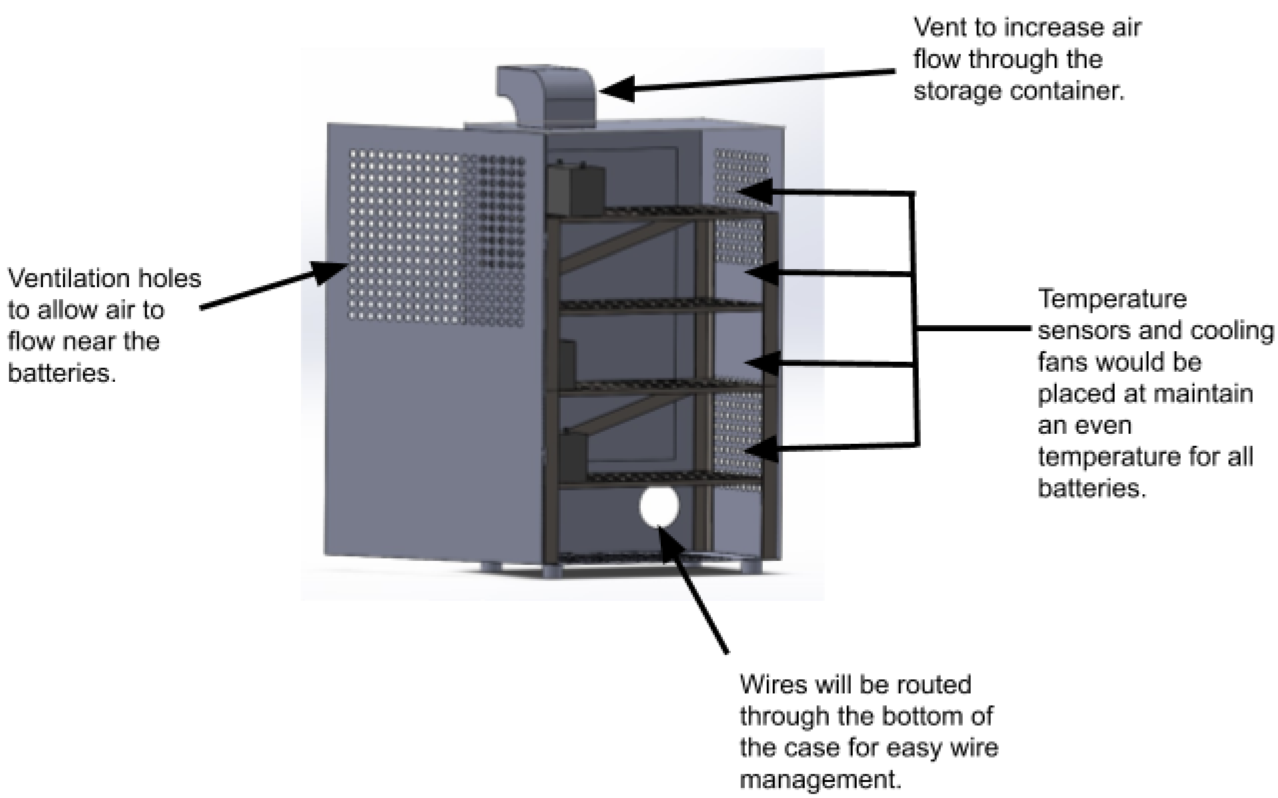


Figure 4: Draft of the battery bank storage case

The distribution boxes will be scaled down to have a current limit of 200A. In order to stay below this current limit, we need to connect the battery bank so that it has two parallel branches. Having the 46 batteries connected in two parallel branches will allow us to meet the distribution box current limit, while also providing the energy required to power the water pump. Equation 2 proves that our battery configuration would meet these requirements.

$$Battery\ Discharge\ Rate\ * \#\ of\ rows\ in\ parallel\ =\ current\ output$$
$$100\ AH * 2\ = 200\ AH$$

$$Battery\ Voltage\ *\ \#\ of\ batteries\ in\ series\ =\ voltage\ output$$
$$12\ V\ *\ 23\ =\ 276\ V$$

$$current\ output\ * voltage\ output\ =\ power\ output$$
$$200\ AH\ * 276\ V\ = 55.2\ kWh$$

Equation 2: Battery bank power output calculation

2.1.2 Circuit Schematic

The microgrid controller makes all the decisions for the overall microgrid, but it still needs to receive accurate data from all nodes within the grid to make such decisions. This is where our circuit enters the equation. It's primary function is to take measurements from the power conversion that takes place between the battery bank and the distribution system, print out relevant data to a display so that someone on-site can monitor the battery bank's status, and send data to the microgrid controller so that it can make its decisions. The next two sections describe the purpose of each component and the critical connections between each component

2.1.2.1 Critical Components Description

The critical component descriptions are as follows.

Microcontroller: ATMEGA328PB-AURCT-ND, Microchip Technology (VA)

The microcontroller is the brain of the PCB and has the most functions. First, reads a signal from the DC/AC converter in charge of power conversion between the battery bank and the distribution box. Initially, we had an external A/D converter which was presented in our design document. However, after that we found that the microcontroller has an internal A/D converter. We decided to remove the external A/D converter and use the one in the microcontroller for two reasons. One, by getting rid of the external A/D converter, we'd save money on components and space on the PCB. Two, it would solve an issue that was brought up during our design review. Our TA brought up a concern that the A/D converter we chose may not function the way we intended because it's output is supposed to drive a 7-segment display, while we chose to use a COG LCD display. Initially, our solution to this problem was to feed the microcontroller the 7-segment data, and internally decode that and match it to characters for display. That is how the microcontroller normally drives displaying characters on a COG LCD display; once the microcontroller know what characters to display, it will send specific bits to the display which internally will decode that to figure out which lights to turn on on the display to make the desired characters. However, this decoding process on the microcontroller side would take up a lot of memory. Therefore, we decided that using the A/D converter on the microcontroller would be the best option, logistically and technically. The A/D converter has an upper limit of 5 volts that can be fed into the analog input voltage pin on the microprocessor. The battery bank can output up to 276 V, so there needs to be a method of reducing this voltage to 5 volts. To implement this, a voltage divider is used. To build the voltage divider, 2 resistors are needed. To minimize the current flowing through this branch, a 10K ohm and 184.5 ohm resistors are used. These resistors are placed in series and the voltage is probed in between them. Equations 3 and 4 show the current drawn by this voltage divider and the voltage measured between the resistors.

$$276V/(10000\ \Omega + 184.5\Omega) = 27.1\ mA$$

Equation 3: current drawn by voltage divider

$$184.5\Omega * 27.1mA = 4.999\ V$$

Equation 4: voltage divider output, signal to A/D converter

Once the A/D converter digitizes this signal, it has to be multiplied by a factor of 55.2 internally, since 276 divided by 5 is 55.2, to calculate the actual voltage output of the battery bank. Once the microcontroller creates a digital number from the analog signal, it moves on to the next two functions.

The second function is to send the digital signal generated by the internal A/D converter to the external display. This data is stored in memory on the microcontroller for the third function, but it is also written to output pins of the microcontroller which are connected to the external display. More on how this data is displayed is covered under the display description.

The last function of the microcontroller is to send this data to the microgrid controller via the ethernet controller. Our microcontroller supports SPI communication and which allows the microcontroller to send data that the ethernet controller can read. This is further explained under the ethernet controller description below. The microcontroller has a good amount of general purpose bi-directional I/O ports that can be programmed for different peripheral functions. For the purposes of this project, and to follow the definitions found in the microcontroller datasheet, port means sets of pins with similar primary and peripheral functions. One of these ports, port B, has four pins with the peripheral function of SPI communications.

These functions fulfill high level requirements #1 and #2.

Ethernet Controller: ENC28J60/SS-ND, Microchip Technology

Information in the Tactical Microgrid System is transferred to and received from the main microcontroller via ethernet. The ethernet controller’s function is to control communication between the microcontroller and the microgrid controller, and to put this data into a format that the microgrid controller can interpret. We can't directly send data from the microcontroller to the microgrid controller because of the differing communication methods, the microcontroller cannot transmit data via ethernet due to its high frequency, which is what the microgrid controller accepts. The ethernet controller will send and receive data from the microcontroller using the SPI communication protocol which the microcontroller can support. It will use that data to create a packet which can be transmitted via ethernet. It can also control the rate at which data is transmitted since it has a control signal that can be toggled to allow or disallow communication between the microcontroller and the microgrid controller. This adds another layer of security that the USACE can utilize if they detect a cyber-security threat on the microgrid controller's side. This fulfills high level requirements #1 and #2.

Ethernet Connector: A121540TR-ND, TE Connectivity AMP Connectors

The ethernet connector is the bridge between the ethernet controller and the microgrid controller. The ethernet controller bridges the gap for communications while the ethernet connector does that physically. The ethernet controller doesn't have a physical ethernet port on it that we can run a cable from directly to the microgrid controller. Instead, the different signals from the ethernet controller have to be routed to the ethernet connector so that an ethernet cable can be used. This fulfills high level requirement #1.

Display: NHD-C0216CIZ-FSW-FBW-3V3, Newhaven Display Intl

The display's primary function is to display the data generated by the A/D converter. Allowing someone physically on-site to see what the output of the battery bank is. This is necessary because of the security measures taken by the USACE towards the microgrid controller. As mentioned earlier, the microgrid controller receives data from all nodes which it uses to make decisions. However, it does not transmit data anywhere on-site pertaining to the status of any nodes; this is all sent to USACE facilities, which are remotely located, and that is the only place where monitoring of the microgrid can take place. This means that if something goes wrong with the microgrid, no one on-site would be able to figure out what the problem is other than physical damage to the exterior of equipment. We wanted to add some sort of monitoring capability on-site to our battery system. Although our display would only show the voltage output of the battery bank, that is enough information for someone to have an idea of the level of charge on the battery bank or use to diagnose a potential problem with power conversion between the battery bank and distribution system. This fulfills high level requirement #3.

Push Button: PA451C1021-134-ND, E-Switch

The push button's function is quite simple. It is supposed to control power flow to the rest of the PCB, allowing the choice of whether or not the battery monitoring PCB is operating or not. This necessity arises in the scenarios when maintenance is being done on the battery bank system.

Consider the scenario where the battery bank is fully charged but not connected to the distribution system. If our circuit is still operating, it would tell the microgrid controller that the battery bank is fully charged. Based on the load demand, the microgrid controller may choose to use the stored energy and turn off the diesel generators which would have been the only other source of power. Suddenly the microgrid and all connected loads would lose power. If instead our circuit was turned off when the battery system was unplugged, the microgrid controller would interpret this as the battery bank is not charged and would make sure to turn the diesel generators on to provide power to the microgrid. Another, secondary purpose is if there needs to be maintenance done on the circuit itself, it would need to be powered off. This fulfills high level requirement #3.

2.1.2.2 Schematic Connection Descriptions

The following describes the connections between components on the PCB, as well as necessary connections to external components. Figure 5 below is a circuit schematic showing the critical connections between components on the PCB and some external components necessary for the circuit to operate.

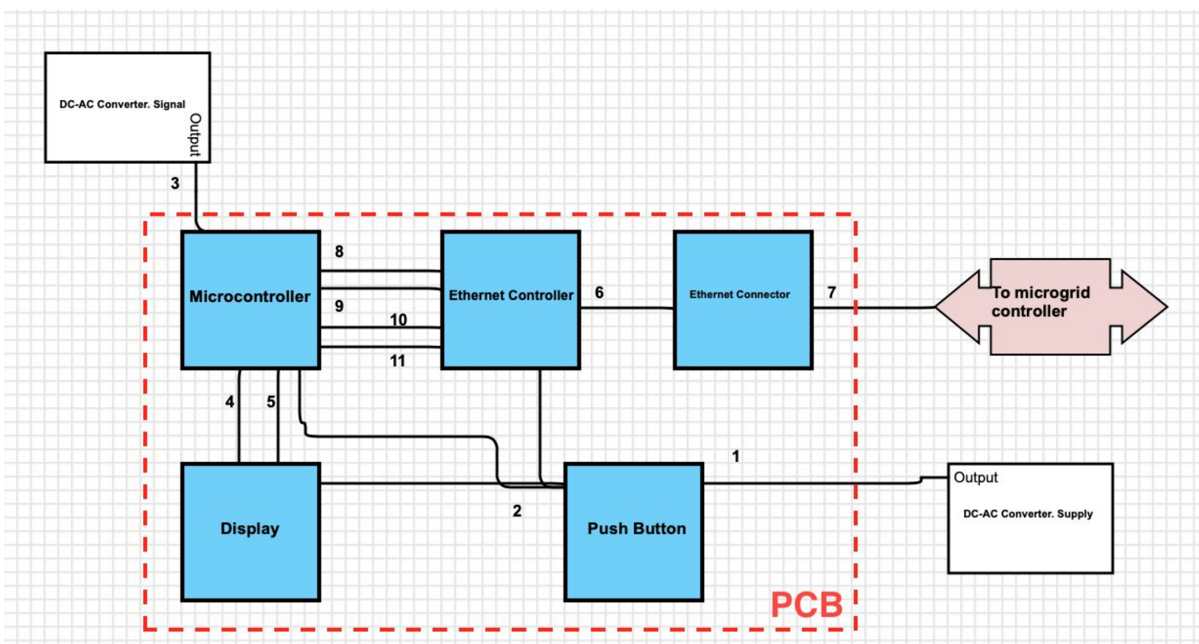


Figure 5: PCB schematic

Each number corresponds to a connection on the circuit schematic, and the description outlines what is connected to what and why. These connections and the corresponding descriptions are limited to the connections between the components on the PCB, not the various connections between pins on individual components because those are standard, generic connections for component operations.

1. This connection is from a DC/AC converter which is needed to convert and step down some voltage coming from the battery bank to supply power to the PCB via a push button.

2. This connection is from the push button to all the supply inputs of the components on the PCB.

3. This connection is from the output of a DC/AC converter to the input of the A/D converter that is built into the microcontroller. The DC/AC converter is used to convert the DC power coming from the battery bank to AC power going to the distribution box. A line from this converter's output will go to the A/D converter inputs on the microcontroller,ADC [7:6]. Specifically these are pins 19 and 22.

4. This connection is from the microcontroller to the lcd display for clock purposes. It is from pin PC1 on the microcontroller to pin 2 on the display, which is the serial clock input on the display. Pin PC1 is selected because it is a clock output on the microcontroller.

5. This connection is from the microcontroller to the lcd display for data purposes. It is from pin PC0, an I/O port on the microcontroller, to pin 3 on the display, which is the serial data input.

6. This connection is an ethernet connection from the ethernet controller to an 8-pin ethernet connector. The 8 pins on the ethernet controller are : pin 3 (CLKOUT), pin 12 (TPIN-), pin 13 (TPIN+), pin 14 (RBIAS), pin 16 (TPOUT-), pin 17, (TPOUT+), pin 23 (OSC1), pin 24 (OSC2).

7. This connection from the ethernet connector to the microgrid controller via an ethernet cable. The purpose of this is so that we can communicate our data with the microgrid controller which can use that to make decisions for the overall microgrid.

8. This connection is between the microcontroller and the ethernet controller. Specifically from port B pin 2 on the microcontroller to pin 9 on the ethernet controller. Port B pin 2 is chosen because one of its peripheral functions is for SPI communication, specifically control of when the other SPI pins can send/receive data. Pin 9 is the control pin on the ethernet controller side.

9. This connection is between the microcontroller and the ethernet controller. Specifically from port B pin 3 on the microcontroller to pin 6 on the ethernet controller. One of port B pin 3's peripheral functions is for SPI communication, specifically it can act as the pin for incoming data in SPI communications. Pin 6 is chosen because it is the pin used to drive data out of the ethernet controller.

10. This connection is between the microcontroller and the ethernet controller. Specifically from port B pin 4 on the microcontroller to pin 7 on the ethernet controller. One of port B pin 4's peripheral functions is for SPI communication, specifically it can act as the pin for driving out data in SPI communications. Pin 7 is chosen because it is the pin used for incoming data sent via SPI communication to the ethernet controller.

11. This connection is between the microcontroller and the ethernet controller. Specifically from port B pin 5 on the microcontroller to pin 8 on the ethernet controller. One of port B pin 5's peripheral functions is for SPI communication, specifically it can act as the clock signal in SPI communications. Pin 8 is chosen because it is the corresponding SPI clock signal pin on the ethernet controller side.

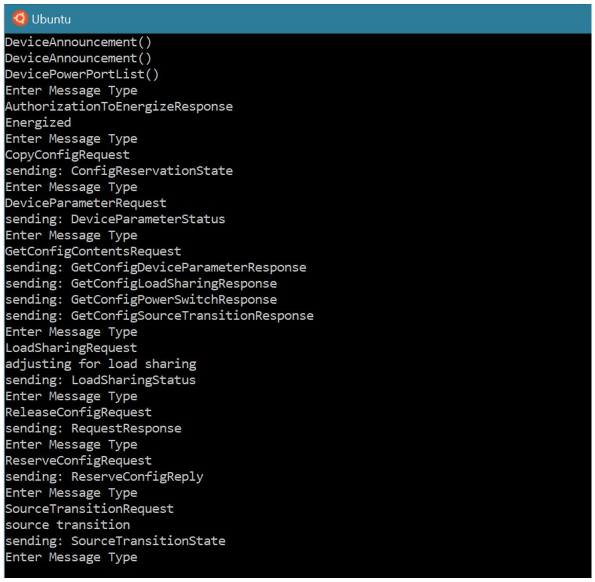
Figure 6 shows the BOM for this circuit. It is well below the budget given to each project group, but it does not take into account the extra resistors and capacitors needed for the components to be fully operational. However, the added cost would still be well within the budget considering the low cost of these passive components is.

Name	Value	Manufacturer	Manufacturer Part Number	Digi-Key Part Number	Quantity	Description
Microcontroller	$1.42	Microchip Technology (VA)	ATMEGA328PB-AUR	ATMEGA328PB-AURCT-ND	1	IC MCU 8BIT 32KB FLASH 32TQFP
Ethernet Controller	$2.79	Microchip Technology	ENC28J60/SS	ENC28J60/SS-ND	1	IC ETHERNET CTRL 8K W/SPI 28SSOP
Display	$11.95	Newhaven Display Intl	NHD-C0216CIZ-FSW-FBW-3V3	NHD-C0216CIZ-FSW-FBW-3V3-ND	1	LCD MOD 32DIG 16X2 TRANSFLCT WHT
Ethernet Connector	$1.39	TE Connectivity AMP Connectors	5406721-1	A121540TR-ND	1	CONN MOD JACK 8P8C R/A UNSHLD
Push Button	$1.55	E-Switch	PA451C1021-134	PA451C1021-134-ND	1	SWITCH PUSHBUTTON SPST 16A 125V
PCB Total cost:	$19.10					

Figure 6: Circuit Bill of Materials

2.1.3 Software Simulation

The TMS-DDS architecture has predefined requirements for what messages our storage system must be able to support. The order of how these messages are sent and received, but not the data inside each message nor how they are executed, are vaguely described in the TMS OMG DDS Implementation Guide. These messages will be generated and interpreted through the software running on our microprocessor, and sent and received through the ethernet adapter. Figure 7. Shows our simulation output, and Figure 8 shows the full list of support messages.



Device Role	Pub/Sub	Topic Name
Storage / STOR	pub	ActiveDiagnostics
	pub	AuthorizationToEnergizeOutcome
	pub	AuthorizationToEnergizeRequest
	pub	ConfigReservationState
	pub	DeviceAnnouncement
	pub	DeviceClockStatus
	pub	DeviceParameterStatus
	pub	DevicePowerMeasurementList
	pub	DevicePowerPortList
	pub	DevicePowerStatusList
	pub	DiscoveredConnectionList
	pub	GetConfigDeviceParameterResponse
	pub	GetConfigLoadSharingResponse
	pub	GetConfigPowerSwitchResponse
	pub	GetConfigSourceTransitionResponse
	pub	LoadSharingStatus
	pub	RequestResponse
	pub	ReserveConfigReply
	pub	SourceTransitionState
	pub	StandardConfigMaster
	sub	AuthorizationToEnergizeResponse
	sub	CopyConfigRequest
	sub	DeviceAnnouncement
	sub	DeviceParameterRequest
	sub	GetConfigContentsRequest
	sub	LoadSharingRequest
	sub	ReleaseConfigRequest
	sub	ReserveConfigRequest
	sub	SourceTransitionRequest

Figure 7: Software Simulation Results Figure 8: Support Message Types

The Storage component, which we will be basing our component off of, is not fully described in the 2.0 Version of the TMS-DDS protocol, which is the version we are referencing. We were unable to get a newer version from the Schweitzer Engineering Lab, who are the developers of the TMS protocol. Due to this, we would have to assume each message's purpose through its name and then deduce the proper response. However, all devices follow a similar start-up protocol, as seen in Figure 9.

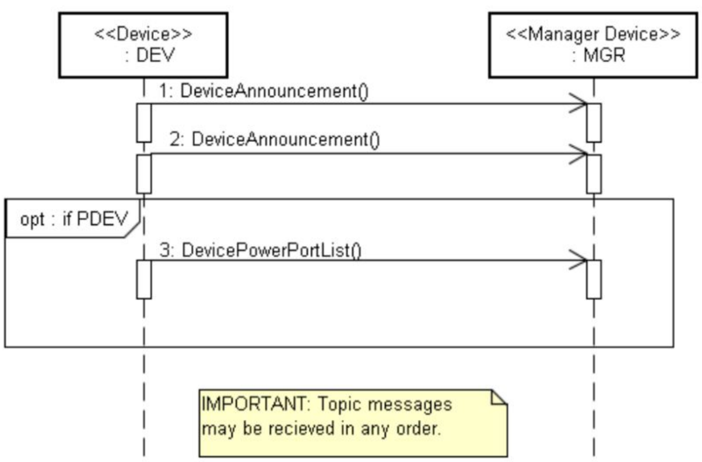


Figure 9: Device-Manager Bootup Communication Protocol

The purpose of this simulation was to better understand the structure of code inside of the microcontroller. By running this experiment, we have learned that the device will need a suitable amount of internal memory to store the structures of each data packet, and it will also need to be able to run multiple threads to send and receive data. This was just an initial simulation, further steps would include writing simulation code to construct full UDP packets, as well as to integrate the batteries' charges. A full printout of the simulation code is in Appendix A.

3 Second Project Conclusions

3.1 Implementation Summary

Battery storage was not a component of the Tactical Microgrid System since the main energy source was diesel generators. The design work done for sizing the battery bank and the types of batteries to use will allow for the use of renewable energy sources, since as solar energy, in the microgrid. Renewable energy sources were the main target to make the Tactical Microgrid System a sustainable solution for civilians where they lack sufficient power infrastructure. Without the exact data of the Guayabota water pumping station, our approach was to make a generalized model so that with more information our calculations could be easily adjusted. Our mathematical model produced an easy method to size the battery bank for different load conditions. Matthew was in charge of the water pumping station analysis and the design of the battery bank.

By creating a circuit schematic outlining the relationship between each component and the physical pin connections, better understanding of each component's functionality was realized and improvements to the design were able to be made. Originally, we had an external A/C converter for measuring purposes, and it was during the schematic building process that we realized the microcontroller we chose already had an internal A/D converter that was better suited for collecting measurements. By making this switch, we were able to reduce memory usage on the microcontroller, and remove extra logic that would have been needed to convert the A/D converter's output to a format that the microcontroller could read. In the end, this also reduced the cost of the PCB as well as the size. Sahil was in charge of building the schematic, and the component and connection descriptions.

One of the critical functional requirements of our project is to send, receive, and interpret various TMS-DDS protocol messages. Pat wrote a high-level simulation software to show how these messages would be relayed. It follows a client-server model, where the client represents our component and the server is the microgrid controller. The simulation is modeled off the protocol specifications in the TMS-DDS implementation guide, although we were not able to receive more detailed documentation about the contents in each UDP packet, we were still able to understand the overall flow of the system, and under which circumstances each message would be sent. This simulation was a valuable learning experience for understanding the role of our device, the activities that would run on our processor, and the work that would be required to deliver a prototype.

3.2 Unknowns, uncertainties, testing needed

Without access to the lab, we are unable to make a physical prototype for testing. At the beginning of the semester we were given the opportunity to test our designs using both the ECE 445 lab and the Construction Engineering Research Laboratory (CERL). We would have used the 445 lab to build the prototype of the communication module and initial testing of the design making use of software developed by the Army Corps of Engineers to mimic TMS-DDS communications. After successful testing of the communication module prototype, we would then test our design at CERL using their model Tactical Microgrid System.

Acquiring the Lithium Ion batteries would have been difficult for testing purposes. The 46 batteries would have cost approximately $20,699.54. Our contact at CERL said they had limited batteries available and ordering additional batteries would have been difficult to do during the semester. We would have scaled down our battery bank to test with the communication module prototype. We also would have used a small scale solar farm to verify that we could model renewable energy sources in the TMS-DDS protocol. Losing access to CERL and the ECE 445 lab eliminating the possibility for physical testing and utilizing CERL's model Tactical Microgrid System.

3.3 Ethics and Safety

The microgrid will be capable of generating high power levels that can cause harm. The diesel generators, solar far and battery bank will be capable of producing over 50 kWh of energy, with currents up to 200A. Our contact at CERL also told us to assume that the individuals in charge of monitoring the microgrid in Guayabota would only receive minimal training. This means that we would have to take great care making sure that microgrid operation is as simple as possible and would require little training. The number one priority for the operation of the microgrid is safety.

The diesel generators add another safety issue, which is the carbon monoxide that would be giving off from burning diesel fuel. The diesel generators need to be properly ventilated to minimize the risk of an individual breathing the toxic fumes. There is also the issue that diesel fuel is extremely flammable, so it will have to be stored in a safe place to make sure it does not get ignited unintentionally.

After the design review, we were made aware of the potential risk of Lithium Ion batteries catching fire in hot climates. The biggest threat is when the batteries would be charging, since that is when the batteries could potentially be pushed into temperature regions that are no longer safe. We designed the battery storage storage case with ventilation and cooling methods in mind, and an additional safety measure that we would add would be temperature sensors to turn off charging or battery usage when the temperature reaches a point where the batteries could potentially catch fire.

An ethical issue that our project will face is the pollution caused by running diesel generators. We have an understanding that diesel engines in this capacity are not sustainable in areas with minimal power grid infrastructure in place, which is why renewable energy sources must be capable for our microgrid so that the diesel energy source will serve the purpose of being a back-up power supply in case of emergency. This is in accordance to the IEEE code of ethics, #1, "...to disclose factors that might endanger the public or the environment." [2].

Another ethical issue that our project will face will be that we must have detailed documentation for safe operation of the microgrid. The assumption that we are under is that the individuals using the microgrid long term will not be an experienced technician. According to the IEEE code of ethics, #5, we are responsible "to improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies…" [2]. This means that we must have clear instructions on how to initially train and operate the microgrid once the system has been deployed, as well as make sure the end user is clear about the purpose of each component and how to safely operate the equipment.

3.4 Project Improvements

The first improvement that we would make to our energy storage system design would be to gather more accurate information on the energy requirements of the water pumping station and the energy output of the solar farm in Guayabota. The COVID-19 pandemic has put the country of Puerto Rico on lockdown, and as a result we had to make multiple assumptions to size the battery bank from various research articles. We kept our equations to estimate the water pump energy output and battery bank size general so that our calculations could easily be adjusted with additional information. Having the necessary information from Guayabota would allow us to fine tune our design to ensure it could be properly deployed in the region.

Another improvement that we would make to our design is consolidating the ac/dc and dc/ac converters in our design. Currently, our design consists of three dc/ac converters, and one ac/dc converter. Further research has shown that there are bidirectional dc/ac converters available that we could utilize in our design. The bidirectional dc/ac converter would connect the battery bank to the distribution box, giving a single path for power to flow based on the battery bank providing energy for the loads or drawing energy for charging. A dc/dc converter would be needed to step down the battery voltage to 5V to power the communication module. The end result would be going from a total of four power converters to only needing two. ANother minor improvement would be to use a voltage transducer to scale down the signal coming from the DC/AC converter which goes to the microprocessor. This would be a more reliable method rather than using a voltage divider. However, voltage transducers are very expensive, at least 3 to 4 times more than our entire project budget.

A third improvement to our design would be to work with the Army Corps of Engineers to redesign the distribution boxes in the Tactical Microgrid System. Their current distribution box design is overengineered, meaning that it has more capabilities than would be realistically used in a civilian application and as a result the cost is too high. The current limit for the design is 800A, and there are many features that would be disabled in prototypes because they were not used. Prior to switching focus to this new project we had brainstormed ideas to decrease the size of the distribution to better suit a civilian environment, and were talking with our contact at CERL to figure out what could be changed with the design. Our preliminary redesign can be found in section 4. Further, detailed, changes would have been needed, including but not limited to: new material research and procurement that is appropriately rated for the new parameters, such as the power bus and the various fuses, and different settings for the relays. A new distribution box design would have made the new Tactical Microgrid System more affordable, and we could have ensured that the distribution box would be capable of incorporating renewable energy sources and additional battery storage.

4 Progress Made on First Project

As mentioned under improvements, one thing that could be improved on is the distribution box. It's over-engineered and the USACE wished to downsize it. At the time of our first design review, we had received documentation on two existing distribution boxes, including the one that the USACE wanted us to redesign. Between then and switching over to the new project, we began a preliminary redesign which meant identifying what components within the distribution box could be removed or replaced. Figure 10 below shows the overall redesign of the 800 A rated box to a 200 A rated one on a one-line diagram level. In green is what would be removed, and red text next to anything in green means a change in rating. The rating changes are necessary since we are taking this down from being rated for 800 A to 200 A. The removed load outputs are there because it was requested that the number of outputs be reduced and that the remaining load outputs be rated for typical civilian applications. A picture of the original one-line diagram can be found in Appendix B.

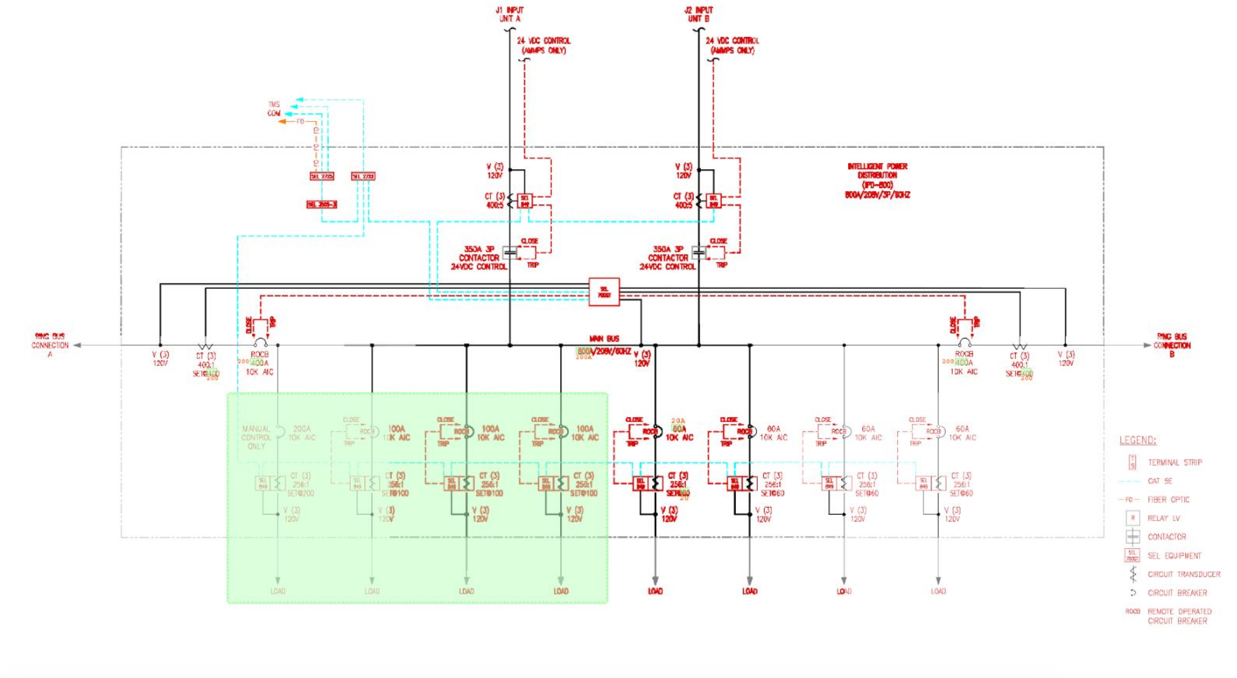


Figure 10: Preliminary redesign of distribution box

5 References

[1] D. Herring. Presentation. “Tactical Microgrid Standard (TMS).” OMG Technical Meeting, Reston, VA, Mar. 19, 2019. Available: https://d2vkrkwbbxbylk.cloudfront.net/sites/default/files/tms-omg-mars-20190319-release-v2_sm.pdf

[2] ieee.org, "IEEE Code of Ethics", 2016. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 13- Feb- 2020].

[3] IEEE Standard for the Specification of Microgrid Controllers," in IEEE Std 2030.7-2017 , vol., no., pp.1-43, 23 April 2018

[4] IEEE Standard for the Testing of Microgrid Controllers," in IEEE Std 2030.8-2018 , vol., no., pp.1-42, 24 Aug. 2018

[5] United States Geological Survey, “Groundwater levels for Puerto Rico,” *National Water Information System: Web Interface*, April 3, 2020. Available: https://nwis.waterdata.usgs.gov/pr/nwis/gwlevels/?site_no=180548065513700&agency_cd=USGS& [Accessed: 3- Apr- 2020].

[6] S. Blount, “Home Water Use in the United States,” National Environmental Education Foundation, 2020. Available: https://www.neefusa.org/weather-and-climate/weather/home-water-use-united-states#Puerto%20Rico [Accessed: 3- Apr- 2020].

[7] B. Peacock, “Energy and Cost Required to Lift or Pressurize Water,” University of California Tulare County, Pub. IG6-96. Available: http://cetulare.ucanr.edu/files/82040.pdf [Accessed 3- Apr- 2020].

[8] Iron Edison, “4.3 kWh Edison Power Plus 24 Volt Complete System,” Iron Edison LLC. Available: https://ironedison.com/edison-power-plus-system-24v [Accessed 4- 15- 2020].

[9] G. Albright, J. Edie, S. Al-Hallaj, “A Comparison of Lead Acid to Lithium-ion in Stationary Storage Applications,” AllCell Technologies LLC, March 2012.. Available: https://www.batterypoweronline.com/wp-content/uploads/2012/07/Lead-acid-white-paper.pdf [Accessed 4- 21- 2020]

Appendix A

```cpp
#include <iostream>
#include <thread>
#include <chrono>
#include <string.h>

using namespace std;

// Not all datatypes are explained in the source material
const char* ACTIVE_DIAGNOSTICS = "ActiveDiagnostics";
const char* AUTHORIZATION_ENERGIZE_OUTCOME = "AuthorizationToEnergizeOutcome";
const char* AUTHORIZATION_ENERGIZE_REQUEST = "AuthorizationToEnergizeRequest";
const char* CONFIG_RESERVATION = "ConfigReservationState";
const char* DEVICE_ANNOUNCEMENT = "DeviceAnnouncement";
const char* DEVICE_CLOCK = "DeviceClockStatus";
const char* DEVICE_PARAMETER = "DeviceParameterStatus";
const char* DEVICE_POWER_MEASUREMENT = "DevicePowerMeasurementList";
const char* DEVICE_POWER_PORT = "DevicePowerPortList";
const char* DEVICE_POWER_STATUS = "DevicePowerStatusList";
const char* DISCOVERED_CONNECTION = "DiscoveredConnectionList";
const char* GET_CONFIG_DEVICE_PARAMETER = "GetConfigDeviceParameterResponse";
const char* GET_CONFIG_LOAD_SHARING = "GetConfigLoadSharingResponse";
const char* GET_CONFIG_POWER_SWITCH = "GetConfigPowerSwitchResponse";
const char* GET_CONFIG_SOURCE = "GetConfigSourceTransitionResponse";
const char* LOAD_SHARING_STATUS = "LoadSharingStatus";
const char* REQUEST_RESPONSE = "RequestResponse";
const char* RESERVE_CONFIG_REPLY = "ReserveConfigReply";
const char* SOURCE_TRANSITION_STATE = "SourceTransitionState";
const char* STANDARD_CONFIG_MASTER = "StandardConfigMaster";

const char* AUTHORIZATION_ENERGIZE_RESPONSE = "AuthorizationToEnergizeResponse";
const char* COPY_CONFIG_REQUEST = "CopyConfigRequest";
const char* DEVICE_PARAMETER_REQUEST = "DeviceParameterRequest";
const char* GET_CONFIG_CONTENTS = "GetConfigContentsRequest";
const char* LOAD_SHARING_REQUEST = "LoadSharingRequest";
const char* RELEASE_CONFIG_REQUEST = "ReleaseConfigRequest";
const char* RESERVE_CONFIG_REQUEST = "ReserveConfigRequest";
const char* SOURCE_TRANSITION_REQUEST = "SourceTransitionRequest";

void sleep(int aTime)
{
    this_thread::sleep_for(chrono::seconds(aTime));
}


void initialize()
{
    cout << "DeviceAnnouncement()" << endl;
    sleep(1);
    cout << "DeviceAnnouncement()" << endl;
    sleep(1);
    cout << "DevicePowerPortList()" << endl;
}


void send(const char* aMsg)
{
    cout << "[Sending] " << aMsg << endl;
}

```

```
void recieve(char* aMsg)
{
    if (strcmp(aMsg, AUTHORIZATION_ENERGIZE_RESPONSE) == 0)
    {
        cout <<  "Energized" << endl;
    }

    else if (strcmp(aMsg, COPY_CONFIG_REQUEST) == 0)
    {
        send(CONFIG_RESERVATION);
    }

    else if (strcmp(aMsg, DEVICE_ANNOUNCEMENT) == 0)
    {
        send(DEVICE_ANNOUNCEMENT);
    }

    else if (strcmp(aMsg, DEVICE_PARAMETER_REQUEST) == 0)
    {
            send(DEVICE_PARAMETER);
    }

    else if (strcmp(aMsg, GET_CONFIG_CONTENTS) == 0)
    {
            send(GET_CONFIG_DEVICE_PARAMETER);
            send(GET_CONFIG_LOAD_SHARING);
            send(GET_CONFIG_POWER_SWITCH);
            send(GET_CONFIG_SOURCE);
    }

    else if (strcmp(aMsg, LOAD_SHARING_REQUEST) == 0)
    {
            cout << "adjusting for load sharing" << endl;
            send(LOAD_SHARING_STATUS);
    }

    else if (strcmp(aMsg, RELEASE_CONFIG_REQUEST) == 0)
    {
            send(REQUEST_RESPONSE);
    }

    else if (strcmp(aMsg, RESERVE_CONFIG_REQUEST) == 0)
    {
            send(RESERVE_CONFIG_REPLY);
    }

    else if (strcmp(aMsg, SOURCE_TRANSITION_REQUEST) == 0)
    {
            cout << "source transition" << endl;
            send(SOURCE_TRANSITION_STATE);
    }
}
```

```
void listen()
{
    while(true)
    {
        char* myMessage = new char;
        cout << "Enter Message Type" << endl;
        cin >> myMessage;
        recieve(myMessage);
    }
}

int main(int argc, char **argv)
{
    thread init(initialize);
    sleep(3);
    thread listener(listen);
    init.join();
    listener.join();
}
```

Appendix B