Water Quality Monitoring

By

Marina Manrique Lopez-Rey

Luis Navarro Velasco

Final Report for ECE 445, Senior Design, Spring 2019

TA: David Null

1 May 2019

Project No. 56

Abstract

The USGS (U.S Geological Survey) has estimated that each person uses up to 130,000 liters of water every year. A big part of this consumption is for drinking and hygiene. Water companies, consumers and environmental scientists need to know the condition of water resources in order to take action if the pollution levels of water raise unexpectedly, given the importance of drinkable water.

The company CERSE (Center for Environment Restoration and Sustainable Energy) is very concerned about this problem, so they want to take action. They created a project that consists on a boat that gets data on water pollution in lakes and rivers, so that the water quality can be monitored. Our goal in this project is to create a solar-powered system capable of uploading that data in real time to a web server, so that data can be accessible in real time.

Contents

1 Introduction

The company CERSE (Center for Environment Restoration and Sustainable Energy) has realized that water pollution is a big issue, and want to make sure that the water that is being consumed is safe. That is why they came up with this project: a boat that analyses water in lakes and rivers and sends real time alerts if water is not drinkable.

Our main aim is to shorten the reaction time that is from when the boat measures the water levels to when the team reads the data and can take measurements to control the levels and alert the population. Right now, the system stores the data on the boat, what makes it impossible to retrieve until it comes to shore. With our project, the team could access the data within 15 minutes of it being read and act consequently.

Our duty in this project is to create some of the modules that this boat needs in order to complete its commitment. We want to provide this boat a GPS system, so that we get information on where the boat is located at every instant of time. This is needed in order to know if the boat is in the position where we want to analyze the water.

In addition, we want to create the module for data transmission system, which is needed to send the data collected by the multiple sensors that the boat is equipped with, and send it to a server so that they can be checked in real time from a laptop or phone. A web page will be created to receive this data and get the alerts that the boat sends depending on the water pollution. In addition to this, we are creating the electric system to power the boat, which includes a solar panel so that the boat can stay offshore for a few days, so that it does not have to pause its job to get charged.

1.1 Main Objectives

1. Data transmission over 3G every 20 minutes for later analysis with web page.
2. Power the electronics of the boat using solar energy, in order to make this project eco-friendly from an energy point of view.
3. Monitoring via GPS with a precision of +/- 2 m in order to track the trajectory of the boat and attach each sensor reading to a set of coordinates.

2 Design

We decided to divide the system into 3 different blocks:

- Power unit
- Control unit
- Communications unit

Breaking up the project into these blocks this allowed us to make an approach in an easier way.

2.1 Block Diagram of the System

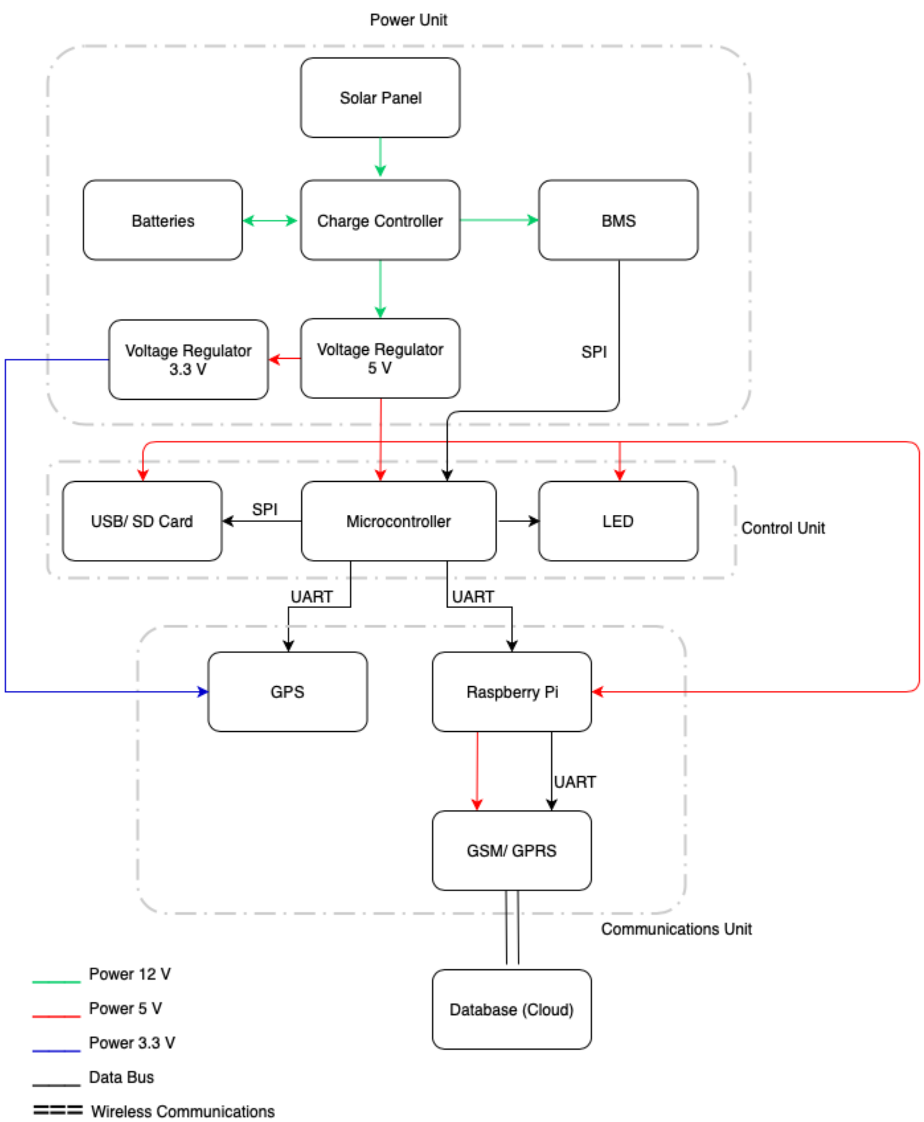

Figure 1: Block diagram

2.2 Communications Unit

2.2.1 Design Procedure

The main challenge when designing the communications unit was to find the layout that offered compatibility between the protocols used by the different modules and chips. Given that the sensors that were provided by CERSE are designed for industrial purposes, they would take great part of the processing capabilities of the ATMega328-PU. Because of this, we decided to move the GSM connection to a Raspberry Pi instead of incorporating it in our PCB. This also gave us the opportunity to use python as our processing language for network connection.

For choosing the different modules, we compared results of people that had worked with the chips and their different outcomes. Also, we tried to find the option that offered the best quality/price relationship for our project.

- The GPS Ublox Neo 6M was chosen because we had previous experience working with this manufacturer and had obtained satisfactory results. Moreover, the software of the manufacturer offers a wide range of configuration options.
- The GSM module SIM800L is one of the most popular modules for this purpose. We chose a 2G module because the power consumption is lower than a 4G. In the United States, T-Mobile is the carrier that provides 2G GSM connection, and so we used a prepaid SIM to connect to this carrier during the demo. If the project was to be used in another country, it should be checked the different characteristics of the cellular network carriers.

The final design steps for the communications unit were the database and web host. We decided to use a MySQL database stored on a remote host so that it could be accessed from the Raspberry Pi with the GSM connection and from the company's computer or other devices. Currently the web is running as a public web page on the internet with user authentication. The language selected for the web page is HTML, with CSS libraries for the user interface design, and PHP for the connection with the database host server.

2.2.2 Schematics

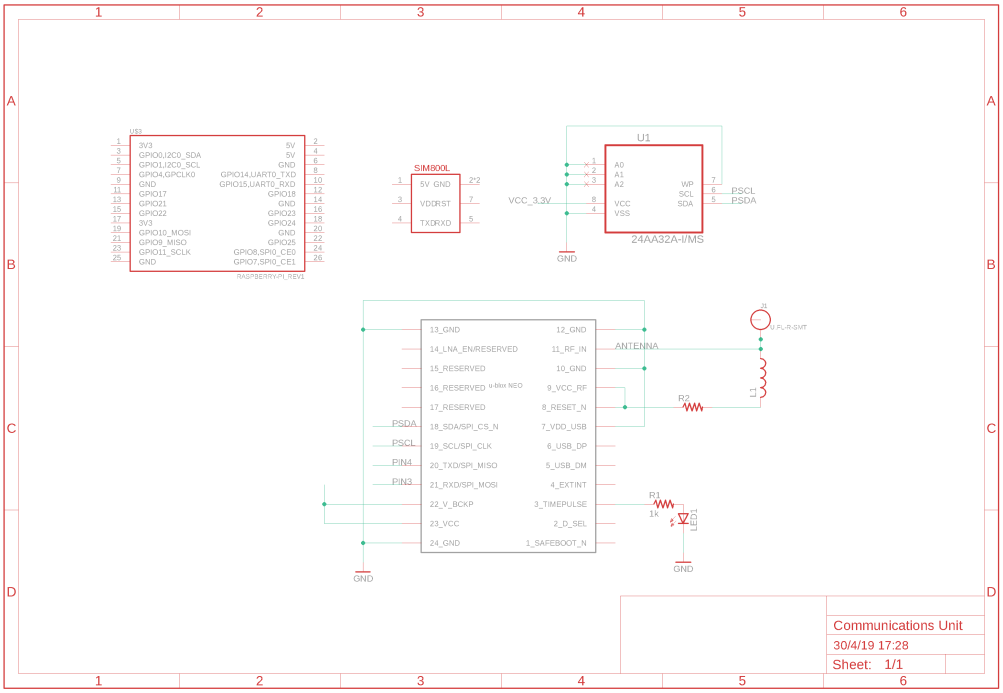

Figure 2: Communication schematics

2.2.3 Description of Components

- GSM/ GPRS: The GSM/ GPRS module is in charge of sending data to the cloud in order for it to be accessible to the users from a computer or android device. It needs to provide 2G/3G data connection and send SMS text alarms when neccessary.

- GPS: The GPS module has two main functions. First of all, data must be classified according to the geographical coordinates and the date in order to keep a clean record. Second, in case data connection is lost, an SMS alarm must be sent with the current GPS coordinates of the device.

- Antenna: An antenna module is necessary in order to increase precision of the GSM and GPS modules, and increase connectivity as well.

- Raspberry Pi 3B+: The Raspberry Pi acts as a connection between the ATMega328 and the GSM module and also as a on board MySQL server.

2.2.4 Data Flow

Figure 3 shows the data flow of the system.

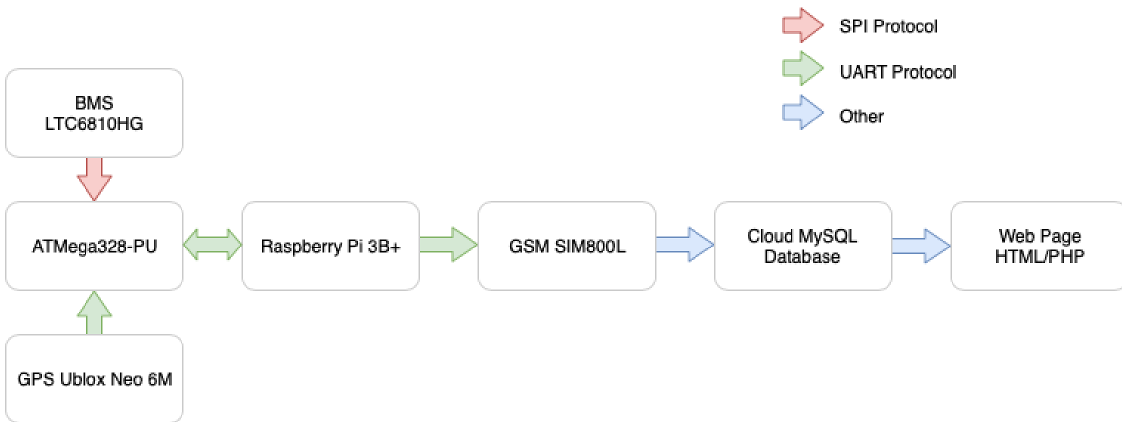

Figure 3: Data Flow

First, the microporcessor receives data from the GPS and BMS via the established protocol for each Pin. Then, it transmits information to the Raspberry Pi following the protocol shown in figure 4. The GSM module is connected to the UART RX and TX pins of the Raspberry Pi and receives the information. Then, GPS and BMS data is transmitted with cellular network protocols to the database host and stored with MySQL commands. Finally, this data is fetched from the host with php scripts.

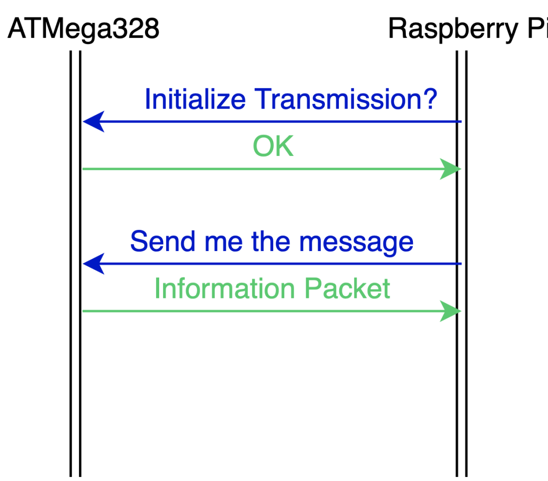

Figure 4: Communication protocol

2.3 Power Unit

2.3.1 Design Procedure

In order to design the full power unit, we needed to have in mind various things:

- Input voltage the load (which is the rest of the system) can have.
- Power consumed by the load.
- Current that the components on the PCB can resist.
- It should be light and not bulky, as it is going to be carried by the boat.
- It is going to be close to water.

First of all, we designed the PCB with the components for the rest of the project, and realised that we needed a 5V voltage regulator, as well as a 3.3V voltage regulator. That 5V voltage regulator is the one that is connected to the power supply (as the 3.3V one is connected to the 5V voltage regulator's output). Having that in mind, we checked that the 5V regulator could get an input of up to 25V. We need to know this value in order to design the power supply.

Now that we have an idea of the voltages that we can input to the system, we start looking for a solar panel. As the power system is going to be on the boat, we need the solar panel to be waterproof (as it could get water spills). We also needed it to be small enough to fit in the boat, and give sufficient power so that it is worth it to add to the project. Following this constraints, we chose a solar panel that was waterproof, is only 33cm long and has a power of 10W (as we estimated our system to have a consumption of 5W).

Also, this solar panel came with a 10A charge controller that outputs 12V. This means that we needed 12V batteries to be able to use the solar panel to charge them. This is why we chose to create battery packs of 3 3.7V batteries (as they are the most common in the market) in order to get 12V battery packs. We were looking for batteries that were not too expensive and that had a good power rate. This is why we chose lithium-ion batteries. We calculated that, in order to be able to power the boat for around 4 days, we should have a total power of 22Ah (as we will show later on in the calculations). This is why we bought 6 batteries of 11Ah and set two packs of these in parallel.

Now that we know the number of batteries that we are using and their power specifications, we have to choose a BMS (Battery Management System). We wanted this component to be able to measure the voltage of the 6 cells and balance the charge between them. The decision of choosing a component that balances the charge between the batteries was taken because the battery system will be in the boat and hard to access to, and in each charge and discharge of the batteries they will not get charged equally. This means that after various charges, some will charge a lot and some will barely get charged. This situation can damage the batteries. Also, we needed it to be able to communicate with the ATmega in order to get data on the battery charge.

In order to optimize the power required by the system, we noticed that we can reduce it a lot by changing the charge controller we are using, or even building our own charge controller. This could be done as future work.

2.3.2 Schematics

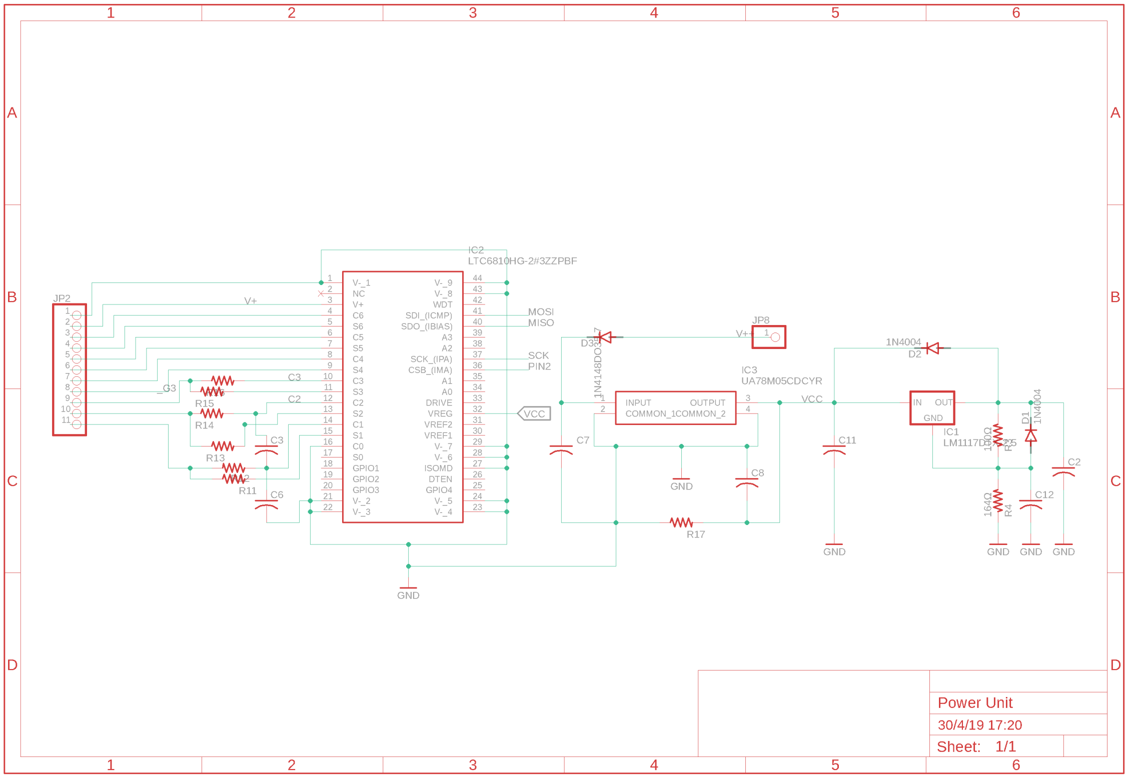

Figure 5: Power circuit schematics

2.3.3 Description of Components

- Solar panel: As we want to make this project eco-friendly, we have decided to use renewable energies to power the electronics of the boat. That is why we are installing a solar panel.
- Batteries: A set of batteries is needed in order to store the energy received by the sunlight and be able to power the electronics. After doing research, we decided to use lithium batteries for this project, as they are a good balance between weight, safeness for the environment and cost. They are the most expensive out of the three types (lead acid, lithium ion and saltwater), but are lighter and have the largest lifespan. They are not as dangerous as the Lead Acid.
- BMS: Battery Management System to control the state of charge of the batteries and check that all of them are working properly.
- Charge Controller: controlls the power flow between the solar panel, the batteries and the rest of the circuit and protects the battery.

2.3.4 Power Calculations

The batteries used for this project are 3.7V and 11 Ah. As we created 2 packs of 3 batteries in series (which means we have a total of 12V per pack) and put them in parallel (which sums the current in the packs), we have a total power of 12V and 22Ah.

As tested after building the project, our system has a total consumption of 0.4Ah. This means that the time that the batteries can power the system before discharging is:

$$Time = \frac{22Ah}{0.4A} = 55h = 2 \text{ days and 7 hours} \tag{1}$$

So, as we can see, the batteries will last for 2 days and 7 hours without being charged. Now, let's calculate how much hours of extra power would the solar panel provide us. In a sunny day, the solar panel chosen for the project can provide 10W of power at 12V. This means that it can provide per hour:

$$Amps = \frac{10Wh}{12V} = 0.833Ah \tag{2}$$

As we can see, the power that the solar panel can provide is equal to the power that the system needs every two hours. Assuming that there are 5 hours of full sun per day in a average day (as it might not be fully sunny), the max power that the solar panel could give us is 4.165A. This power is sufficient to power the system for:

$$\text{Extra time} = \frac{4.165A}{0.4A/h} = 10.41h \tag{3}$$

This means that the batteries would supply power for 13.59 hours a day, while the solar panel would provide the sufficient power during 10.41 hours a day. This means that the total time we will be able to power the system before charging the batteries is of:

$$Days = \frac{55h}{13.59h/day} = 4 \text{ days before charging batteries} \tag{4}$$

2.4 Control Unit

The control unit manages every other module of the system. Receives the data from the sensors installed on the boat and prepares the information to be sent over the GSM module, as well as stored in an SD. It also provides feedback or alarms over the LED module.

2.4.1 Design Procedure

For the control unit we decided to use the microcontroller ATMega328-PU because it offered SPI, UART and I2C protocols to connect to the rest of our components. Moreover, we decided to install the Arduino bootloader so we could use libraries created for this platform. One of the main libraries was "TinyGPS", that helps with the connection of the GPS module and the ATMega328-PU. In addition, we decided to include an SD Card as a backup system in case the network connection of the system was lost. This last one was not a critical component of our system but could be useful in the future.

As far as protocols are concerned, we decided to communicate with the BMS via SPI and the GPS with UART because these were the recommended protocols by the manufacturer. On the other hand, we designed an I2C and an UART protocol for the communication with the Raspberry Pi, but decided to use the latter due to buffer overlap.

2.4.2 Schematics

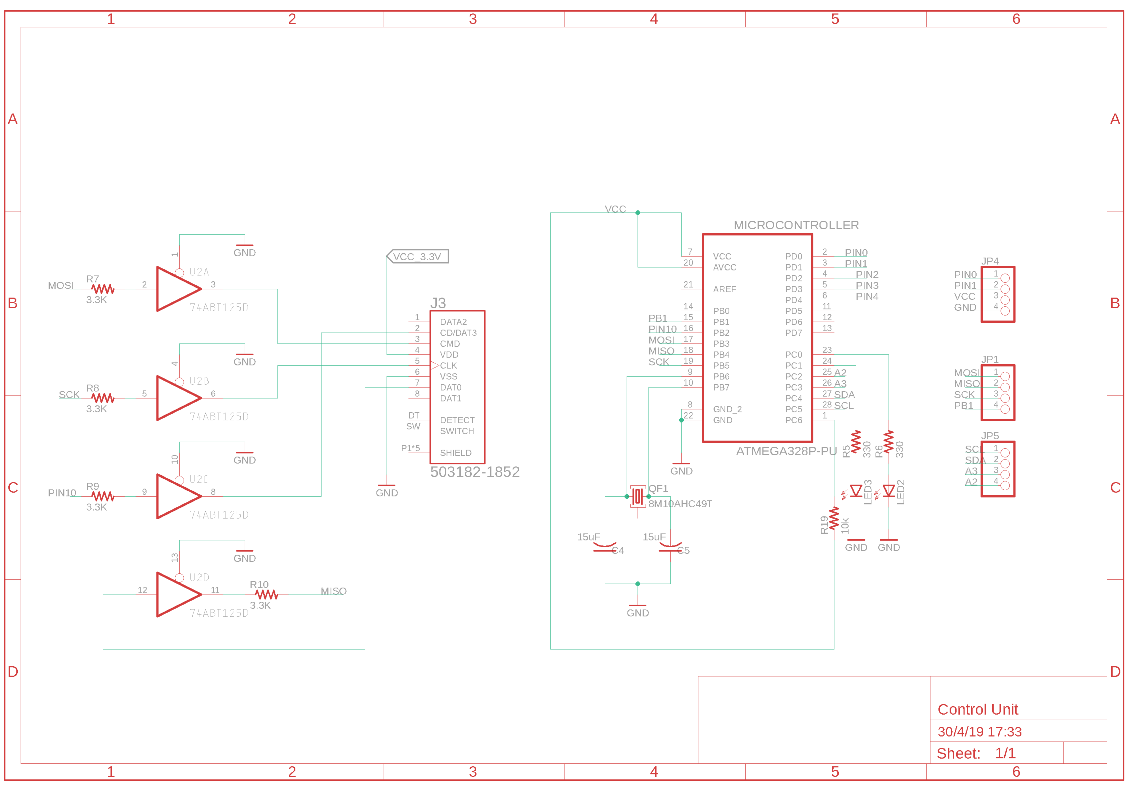

Figure 6: Control circuit schematics

2.4.3 Description of Components

- Microcontroller: The microcontroller handles all the data. First, it collects the data from the sensors via its connection pins. Then it stores the data in the SD/ USB module via SPI connection and sends the data to the Raspberry Pi via UART. We will work with ATmega328, a model that must be able to process the great quantity of data that would be generated by the sensors.
- LED: Two LEDs in the protoboard provide information about the communication status.
- SD Module: The SD Card module will be connected to the microcontroller via SPI. This is necessary because given the case were data connection is lost, the GSM module wont be able to send the data, and we need it to be stored in a SD module to avoid losing the data.

2.5 PCB Design

Figure 7 shows the final board design for the board that is the core of our project. The PCB is formed by:

- Control Unit: Microcontroller, SD Card reader and status LED.
- GPS chip and its necessary connections.
- Battery Management Control (BMS)
- 5 V Voltage Regulator (UA78M05CDCYR)
- 3.3 V Voltage Regulator (LM1117D)

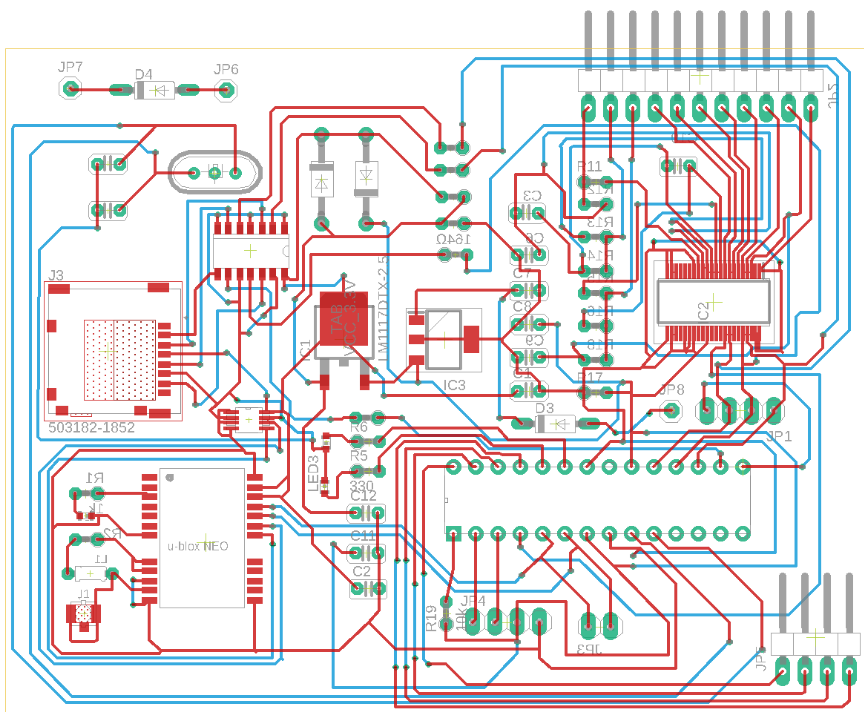

Figure 7: PCB design

3 Design Verification

As we can observe in Table 3: Requirements and Verifications (Appendix A), all our requirements but one were completed for the demo of this project. In this section, we will analyze how the verifications of each unit were developed.

3.1 Communication Unit

To satisfy the requirements shown in Appendix A for this module, we checked if the Web application showed the data that was being sent by the GPS and the BMS. In order to prove this, we needed to take some steps for the verification of each module.

For the verification of the GPS modules, we used Google Maps in order to mark the different coordinates the sensor was giving while our system was moving. We obtained the accuracy needed, as can be shown in figure 8.

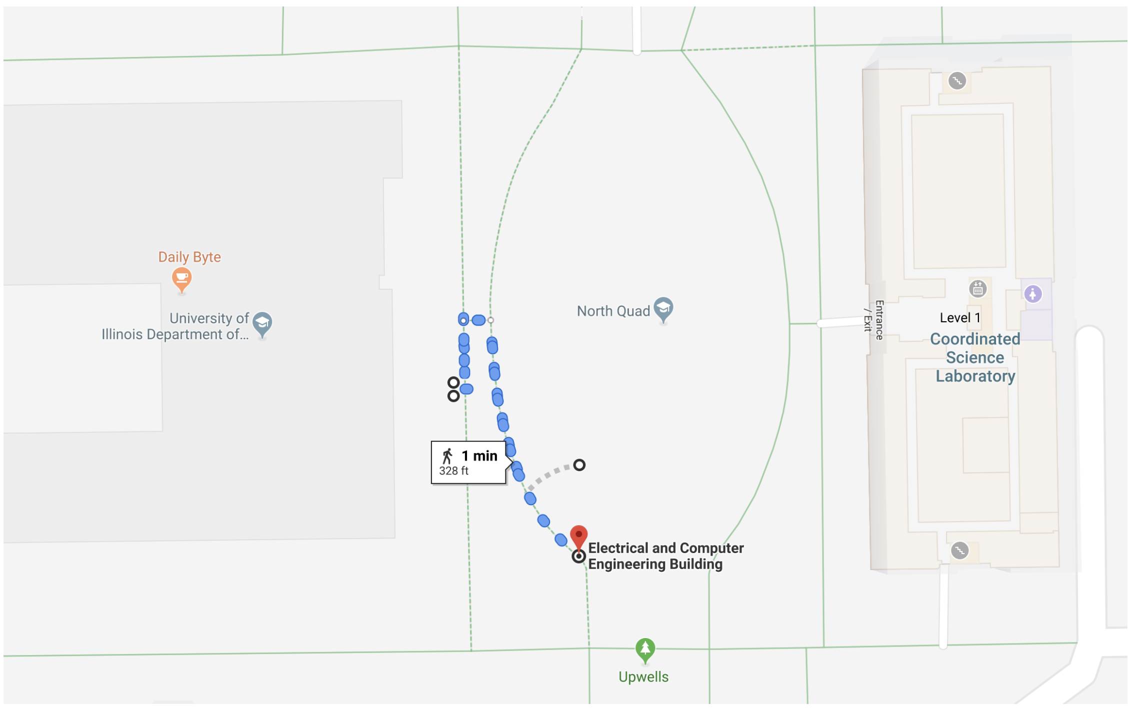

Figure 8: Test results visualized in Google Maps

The data of the GPS was later sent to the microcontroller. We could see that the data was received correctly by monitoring our PCB via a Serial Monitor. The following step was to send the data from the ATMega328 to the Raspberry Pi, and connect the entire system to a network via a GSM module. We designed a protocol based on the TCP handshake in order to communicate the PCB and the Raspberry Pi. After doing some trials, the success rate of the communication was 100% given that no packets were lost and all the data was received correctly.

The final step was to send data from the system to the server using the GSM module. To verify the proper operation of the GSM, we first verified the connection using AT commands and sending an SMS to our phone. The system used a 2G SIM Card and connected to T-Mobile network. After that, we established

3G connection with a python program we designed, and could check that the data sent was the same that appeared in our database. Finally, that data was showed in the Web application in tables as well as in graphs.

3.2 Power Unit

In order to test the power unit, we connected the whole power system (solar panel, batteries, charge controller and the rest of the project) and located the solar panel under the sun. We could see that the project was powered at all times, as we were getting data in our web. Also, thanks to the LCD screen that the charge controller is equipped with, we could confirm that the voltage in the batteries was increasing, which means that they were being charged.

In order to check that the power was being provided properly to each component, we had to check that the output of the two voltage regulators we were using was how it should be. This was checked why the system was connected as mentioned before, and using a voltmeter to measure the output in both voltage regulators.

3.3 Control Unit

The proper functioning of the Control Unit is critical for the rest of the system. As we mentioned before, the ATMega328 received without any problem the data from the batteries and the GPS. Such measurements, were compared to the data read by the charge controller and the GPS manufacturer software.

4 Cost

4.1 Parts

Table 1: Parts Costs

Part	Quantity	Unit Cost ($)	Totall Cost ($)
GPS Neo-6M	1	15.00	15.00
GSM SIM800L SIM Com	1	9.00	9.00
Microcontroller ATMega328-PU	1	2.00	2.00
Raspberry Pi 3B+	1	39.99	39.99
SD socket	1	2.23	2.23
Quad Buffer 74ABT125D	1	0.60	0.60
Voltage Regulator LM1117D	1	1.14	1.14
Voltage Regulator UA78M05CDCYR	1	0.62	0.62
BMS LTC6810HG	1	16.04	16.04
Solar Panel and charge controller	1	33.66	33.66
Battery	6	1.97	11.83
Battery holder	4	1.75	6.99
Total	-	-	139.1

4.2 Labor

Table 2: Labor Costs

Function	Labour Hors (h)	Totall Cost ($)
PCB Design	25	300
Battery Design	20	240
Web Development	30	360
Database Configuration	20	240
GPS Calibration	20	240
GSM Configuration	30	360
Charger and Solar panel study	30	360
System Integration	40	480
Total	215	2580

5 Conclusion

5.1 Accomplishments

We achieved a complete functional system that was able to meet all its requirements and objectives for this project. On the communications side, data was read from the GPS module and the batteries by the ATMega and sent via UART to the Raspberry Pi. With the GSM module, we could connect to our server and upload data to the database. A webpage was designed and deployed in a public domain so that the data and the graphs could be accessed from different platforms. On the Power side, the solar panel and batteries supported the entire system and provided all the voltage and current needed. The ATMega328 received its power from the 5 V regulator and the rest of the PCB components from the 3.3 V regulator, that gave a real value of 3.4 V. In figure 10 we see the user interface of the web application, and in figures 9 and 10 respectively, the final layout of the PCB and the complete system.

The project was fully operational on demo day except for minor components that were not critical to the functionalities or goals of the project.

The SD card module was not operational during the demo. Even though the SD card was detected from the socket of our board by the ATMega, no data was written. Further debugging needs to be done given that the code uploaded was correct. In order to preserve an on board storage and give the Raspberry Pi more functionalities, a MySQL server was installed on localhost to store the data.

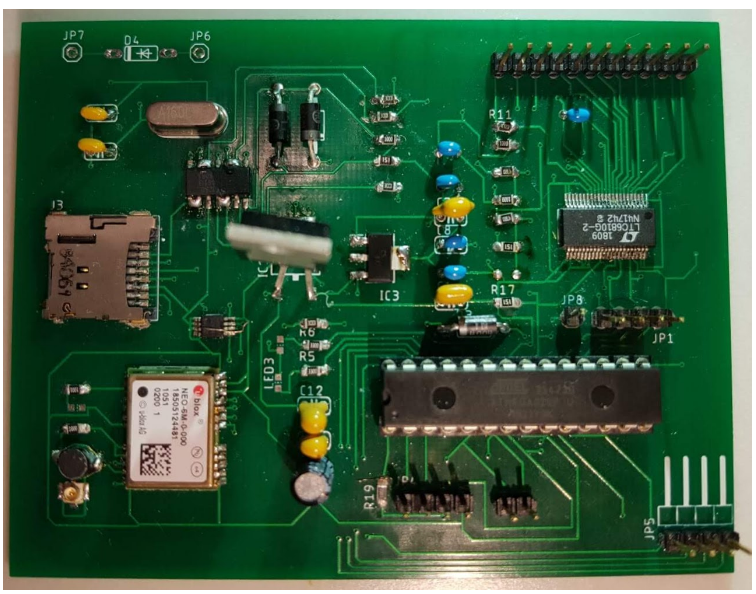

Figure 9: PCB

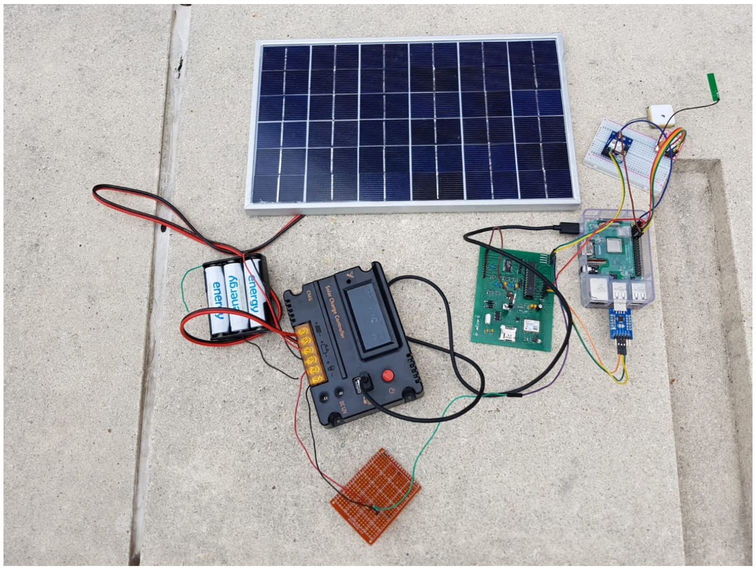

Figure 10: Integrated system

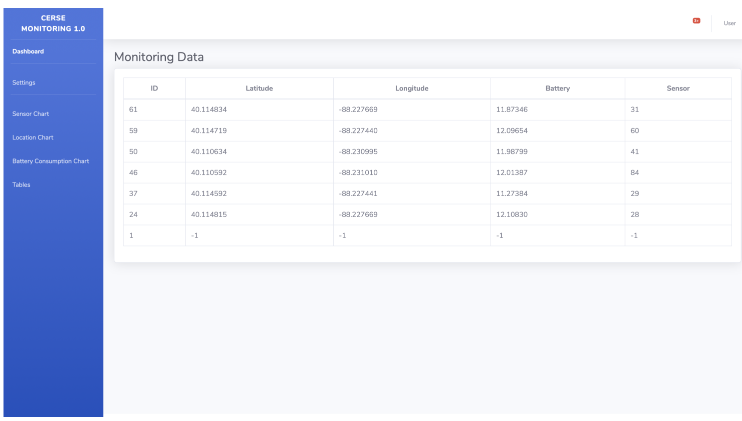

Figure 11: Designed web

5.2 Uncertainties

The most remarkable uncertainty about our project is related to one of our two GPS chips, given that the GPS of the PCB was not fully configured. Although the requirement for GPS data was met by an external module connected to our PCB, the GPS chip inserted in the PCB was not receiving satellite information.

Even though the chip was detected by the manufacturer software and configured according to the correct parameters, it wasn't able to lock connection with any satellites. This lead us to the conclusion that it was not a problem of the circuit design but of one of its components, because given the other situation, the manufacturer's software would not detect the chip.

Further debugging lead us to the belief that the problem could rely on the storage of the configuration set or the connection to the EEPROM of the board. However, this needs a thorough investigation.

5.3 Ethical considerations

This system should entail no risk to the environment, or to the animals living in or around the lakes or rivers that are being analysed. Also, if the water being analysed is polluted, the consumers of this water should be alerted instantly.

In addition, when designing this project, we need to keep in mind the environmental impacts that it could have and also guarantee a sustainable system given CERSEs commitment to sustainable practices and its commitment to the environment.

According to the IEEE Code of Ethics, the project we need to agree to be honest and realistic in stating claims or estimates based on available data. This point has great importance to our project given that the data collected must be analyzed by not biased algorithms and keep peoples interests and not companies first, given that drinkable water is a priority in our society. Information should be managed and published in a honest way, always having in mind that peoples health is in risk.

The boat should be perfectly sealed, so that no water gets into the electric system and entails no risk. If water gets into the system, there could be a short circuit. This could destroy the electric system and could be dangerous for the animals that live in the water that is being analysed, as they could get an electric shock. This is also a reason for choosing a waterproof solar panel.

It should be ensured that that the electric system is completely isolated from the body of the boat. If there is not a complete isolation, the boat could be electrically charged and could be very dangerous for the living creatures of the lake or river, as they could get an electric shock.

The chosen motor should be an electric motor. These type of motors are the most eco-friendly ones. Using a petrol or gas motor would pollute the water, as oil could get spilled in the water, and also the smoke it produces is dangerous for the environment. Also, this smoke and oil could taint the water samples that the boat collects.

We should have in mind that if there is an issue with the boat and the batteries get into the water, they should be collected immediately. These batteries are made of very polluting and dangerous chemical materials, such as lithium or lead. This is why they should be collected from underwater if the boat sinks, before they damage the water.

5.4 Future work

We are currently working with CERSE to be able to incorporate our system into their project. Given that for our project we did not use any of their sensors, it is now needed to create a communication channel between the two parts of the project. This would be achieved by creating a serial communication between the sensor hub and the microcontroller of our PCB (ATMega328-PU) However, this is still a proposal and needs further research and testing.

Once the connection to the sensors is completed, we would like to increase the project's AI capabilities, since one of the ideas we envisioned was to be able to analyze the historical data and use Big Data algorithms in order to process the measurements faster and set precise alarms.

In the Communications Unit, in order to increase reliability, a bit parity or checksum verification is needed between the ATMega328 and Raspberry Pi transmission. Also, we could strengthen the security measures, so that data such as location coordinates is not transmitted as plain text between the system and the web server. This could be achieved by creating encrypting the data via hash (md5 or SHA256) or using public/private key verification.

In the Power Unit, new calculations would need to be made once the project joins the sensors to estimate the lifespan of the batteries in the new conditions and establish if any modification is needed.

As to PCB layout , one proposal would be to configure the voltage regulators in parallel in order to increase efficiency of the regulators. In this case, we would need to switch the 3.3 V regulator in order for it to expand its input voltage range.

References

[1] *Adafruit Industries LLC 254, DigiKey Electronics, February 21 2019 [Online]:* https://www.digikey.com/product-detail/en/adafruit-industries-llc/254/1528-1462-ND/5761230?gclid=EAIaIQobChMI8Ku9kZzM4AIVdBtBh2BAQ4TEAYYAyABEgKgKfD$_{B}wE$

[2] *GNSS 5 click, MikroElektronika, February 21 2019 [Online]:* https://www.mikroe.com/gnss-5-click

[3] *GSM 4 click, MikroElektronika, February 21 2019 [Online]:* https://www.mikroe.com/gsm-4-click

[4] *NEO-M8 Datasheet, U-blox, August 5 2016 [Online]:* https://www.u-blox.com/sites/default/files/NEO-M8-FW3$_{D}ataSheet_{2}8UBX - 1503108629$.*pdf*

[5] *Positive Voltage Regulators, Texas Instruments, May 2003 [Online]:* https://www.sparkfun.com/datasheets/Components/LM7805.pdf

[6] *SARA-G3 series, U-blox, March 27 2018 [Online]:* https://www.u-blox.com/sites/default/files/SARA-G3$_{D}ataSheet_{2}8UBX - 1300099329$.*pdf*

[7] *How to choose the best battery for a solar energy system, Energysage, January 1 2019 [Online]:* https://www.energysage.com/solar/solar-energy-storage/what-are-the-best-batteriesfor- solar-panels/

[8] *Lithium-Ion Battery Management Systems and large battery packs, Li-Ion BMS, [Online]:* http://liionbms.com/php/index.php

[9] *Como elegir bateras para paneles solares, Calefaccin solar, February 14 2019 [Online]:* http://calefaccion-solar.com/como-elegir-baterias-para-paneles-solares.html

Appendix A Requirement and Verification Table

Table 3: System Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Requirement: GPS (a) Location error must not be over 2m. (b) Must transmit latitude and longitude coordinates of the boat. (c) Must send data to the micro controller.	1. Verifications (a) Move the system from a known point to a known destination and measure the distance error. (b) Study the data store in the web server. (c) Control, via SD card or an auxiliary LCD, if the micro controller is able to receive its location data.	Y
2. Requirement: GSM (a) Connect to the server and send data. (b) Connect to the network with the highest speed (2G/3G/4G) to minimize latency.	2. Verifications (a) Log into the server and check that it is receiving the coordinates of the boat. (b) Send SMS to designated number to test connectivity 2G and email to test 3G connectivity.	Y
3. Requirement: Microcontroller (a) Show alerts if any of the modules stops working. (b) Configure the micro controller and GSM module to connect to the server every 20 minutes and send the sensor data file. (c) Receive position coordinates from the GPS sensor and match each sensor reading with its GPS coordinates.	3. Verifications (a) Code by color LEDs to verify the proper performance of the microcontroller. (b) External software may be needed to measure the packets that the system is sending to the database system (Wireshark software) (c) Check every sensor data on the server is linked to a latitude and a longitude coordinate.	Y
4. Requirement: SD Card (a) Receive and log data from the microcontroller	4. Verifications (a) Send a sample.txt file to the micro controller and compare this file with the one saved on the SD card.	N

Continued on next page

Table 3 – continued from previous page

Requirement	Verification	Verification status (Y or N)
5. Requirement: Batteries (a) Power the electronic components of the project. (b) Be able to be charged by solar panel.	5. Verifications (a) Place a load equal to the average power consumption of the whole system and check if the batteries can power this equivalent system. (b) Discharge the batteries and connect them to the solar panel. Then check that the battery load has increased by discharging the batteries again.	Y
6. Requirement: Web Application (a) Retrieve and process data from database or web service. (b) Graphical representation of the data retrieved from the server. (c) Show data with an actualization delay of at most 15 minutes.	6. Verifications (a) Compare data from application with data stored in the server. (b) Represent sample data in a processing software (MATLAB) and compare the accuracy of the desktop application. (c) Check real time timestamps of the data from the server.	Y
7. Requirement: Solar Panel (a) Charge the batteries.	7. Verifications (a) Fully discharge the batteries and then charge them, making sure that the solar panel is powering them.	Y
8. Requirement: BMS (a) Be able to analyze the state of charge of the battery.	8. Verifications (a) Check if the microcontroller receives the battery charge data from the BMS.	Y