High altitude soil temperature/moisture monitoring

Project Proposal and Bill of Materials

October 30, 2019

Team: Samuel Croft, Carson Fiechtner, and Tyler Vendetti

Mentor: Dr. Kevin Negus

Sponsor: Dr. Martha Apple

Contents

[1. Problem Statement 2](#_Toc22996907)

[1.1 Introduction 2](#_Toc22996908)

[1.2 Background 2](#_Toc22996909)

[2. Research 2](#_Toc22996910)

[2.1 Components 2](#_Toc22996911)

[2.1.1 SHT-10 Soil Temperature/Moisture Sensors 3](#_Toc22996912)

[2.1.2 Arduino Uno Rev 3 3](#_Toc22996913)

[2.1.3 RockBLOCK 4](#_Toc22996914)

[2.1.4 Battery 5](#_Toc22996915)

[2.2 Component Interoperability 5](#_Toc22996916)

[2.3 Power Considerations 5](#_Toc22996917)

[2.4 Code 5](#_Toc22996918)

[4. Deliverables 5](#_Toc22996919)

[4.1 Prototype 5](#_Toc22996920)

[4.2 Schedule 5](#_Toc22996921)

[4.2 Block Chart 6](#_Toc22996922)

[5. Bill of Materials 6](#_Toc22996923)

[6. Conclusion 6](#_Toc22996924)

[7. References and Appendix 6](#_Toc22996925)

[7.1 References 6](#_Toc22996926)

[7.2 Appendix 7](#_Toc22996927)

[A. Bill of Materials 7](#_Toc22996928)

[E. Schedule Gantt Chart 8](#_Toc22996929)

[F. Project Overview Block Chart 9](#_Toc22996930)

# 1. Problem Statement

## 1.1 Introduction

Dr. Martha Apple is looking for ways to improve her high-altitude soil temperature monitoring setup at Goat Flat in the Pintler Mountains. Presently, Dr. Apple uses several HOBO Tidbit V2 Temperature Sensors [1], all of which log their own data individually. When Dr. Apple would like to retrieve that data, she must hike up to Goat Flat, locate each sensor through line-of-sight pictures and painted rocks, transfer the data to her ruggedized computer, and hike back down.

While there are several things that Dr. Apple would like to improve on, our team has decided to focus on retrieving the sensor data remotely, removing the need for as many trips up to Goat Flat. Specifics on how we plan to implement remote communication can be found in the Research and Deliverables sections below.

## 1.2 Background

As was mentioned earlier, Dr. Apple had multiple ideas for how to improve her high-altitude monitoring setup. Specifically, she had three upgrades in mind, ranked in order of importance to Dr. Apple:

1. Add soil moisture monitoring capabilities

2. Remotely retrieve sensor data

3. Easier retrieval of temperature sensors

While we decided to focus on remote retrieval of sensor data, we did not overlook the others. As you will see later in the proposal, we have decided to use sensors that monitor both soil temperature and moisture, removing the need for separate sensors.

We also thought of a few ways to make finding the sensors easier. Since our sensors will send data remotely, there will be much less need to find the sensors on a regular basis. In the case of a sensor malfunction or dead battery, the sensors can be found by simply following the wire that leads from the data hub. We can also include ring magnets, which would make finding the sensors or data hub with a metal detector straightforward.

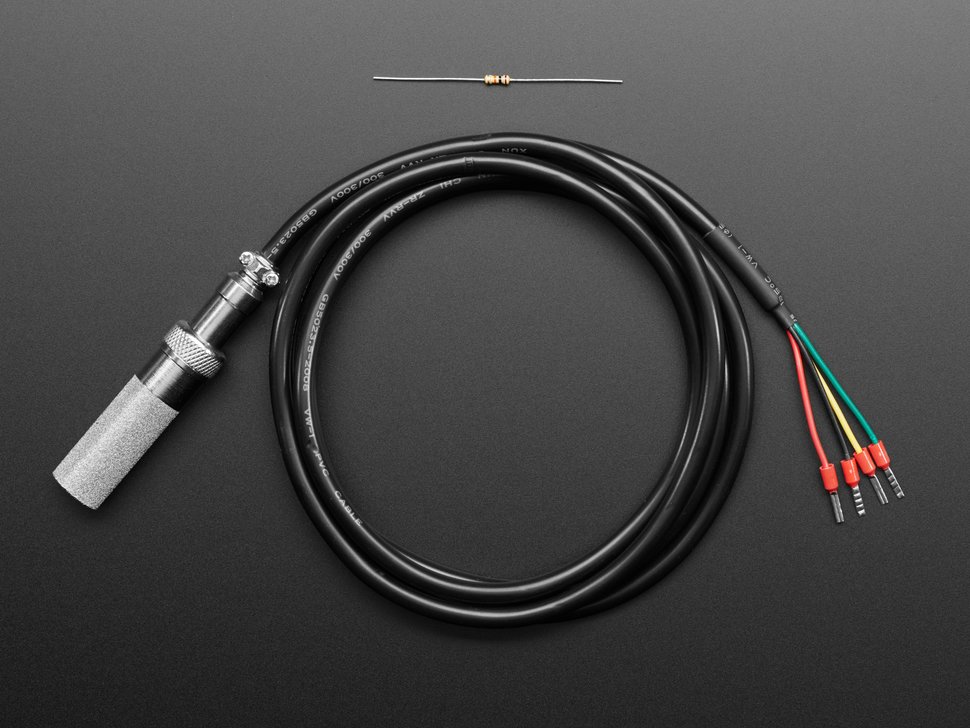
# 2. Research

## 2.1 Components

In thinking about how we can create a great project that is achievable, we’ve strived to keep all aspects of the design as simple as possible. We’ve chosen wired sensors that are built to interface with a microcontroller [2]. We’ve chosen a power-efficient microcontroller [3] to ensure a long useful life. Perhaps most importantly, we’ve chosen a communication modem that is designed to work in the harshest, most remote locations and makes messaging through a microcontroller simple [4]. Details on each individual component of the system, as well as alternatives that were not chosen, can be found below as well as in their respective datasheets in the attached appendix.

### 2.1.1 SHT-10 Soil Temperature/Moisture Sensors

We chose to use wired SHT-10 combined soil temperature and moisture sensors for several reasons. First, using these sensors allows us to add soil moisture monitoring [2], which was our highest-priority request from Dr. Apple. Second, because these sensors are wired and not wireless, we believe they will be more reliable and less likely to unexpectedly fail or lose connection to their host. Finally, they even have a code library built specifically for our microcontroller, the Arduino, that should make integration relatively painless [2].



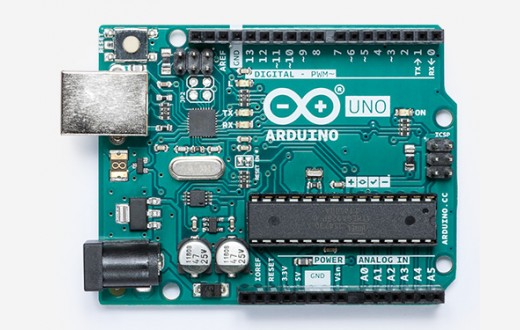
SHT-10 Soil Temperature/Moisture Sensor [2]

We considered numerous alternative sensors while making our decision. [TODO: Talk about wireless sensors].

[Insert images of wireless sensors considered]

### 2.1.2 Arduino Uno Rev 3

Because we had a few frontrunners in the microcontroller space, we didn’t consider as many microcontrollers as we did soil sensors. Early in the design of our project, we had expected to use a Raspberry Pi as the microcontroller. However, once we discovered that Raspberry Pi’s don’t include a sleep mode [5], we were forced to pivot. The Arduino was the next clear choice; it included a sleep mode and advertised interoperability with both our chosen sensors and modem [3]. It also allows for the temperature sensors to be multiplexed, which greatly increases the scalability of our system [6].



Arduino Uno Rev 3 [3]

### 2.1.3 RockBLOCK

The RockBLOCK was the standout choice in the decision for a communication modem. Built to make interfacing with a microcontroller easy, the RockBLOCK uses an Iridium modem to allow for satellite communication from anywhere in the world with a view of the sky [4]. It is also cheaper than the alternatives we had in mind [4][7][8], which made the choice that much easier.



RockBLOCK Iridium SatComm Module [4]

Before we discovered the RockBLOCK, we were considering modems from both Iridium and Globalstar. Iridium is designed to work world-wide and offers a starter kit for those new to satellite modems but was the most expensive choice and left communication setup with a microcontroller unclear [6]. Globalstar was less expensive but required a considerable amount of code before it could work and didn’t operate as efficiently in southern latitudes [8].

Iridium Edge Demo Kit [7] Globalstar STX3 Satellite Transmitter [8]

### 2.1.4 Battery

TODO

## 2.2 Component Interoperability

TODO

## 2.3 Power Considerations

TODO

## 2.4 Code

Fortunately, much of the code required for this project already exists and is available to the general public. Code that allows the Arduino to read data from the SHT-10 sensors is available in a Github repository [9], and RockBLOCK provides developer guides and an API to help developers send messages through a microcontroller [10]. There is even a whole blog post detailing how to put an Arduino in sleep mode [11]. There will undoubtedly be some modifications required to make everything work cohesively, but it is encouraging that the majority of the code we need is readily available.

# 4. Deliverables

## 4.1 Prototype

The goal of this project is to deliver a prototype system that can remotely deliver sensor data to an email address or IP address. This prototype will be tested in the wilderness surrounding Montana Tech’s campus to simulate the weather conditions of Goat Flat.

It is important to note that we do not expect our project to work constantly. Because our modem requires a view of the sky and the Pintlers are covered in snow for a majority of the year, it is likely that our project will only be able to transmit data during the few warmest months every year.

## 4.2 Schedule

A Gantt chart has been included in Appendix D to give a rough estimation for the timeline of our project. Please keep in mind that these dates are estimations and will likely change throughout the project.

## 4.2 Block Chart

A block chart has been included in Appendix E to provide a general overview of what our final project might look like.

# 5. Bill of Materials

A bill of materials has been included in Appendix A. The total estimated cost of the project is $794.91. Most of that cost is associated with the RockBLOCK Satellite Modem and its operation. Along with purchasing the unit, we must also pay for monthly line usage and credits to send messages. The next highest cost is the temperature sensors, mostly because we need multiple sensors to create a representative prototype. The remainder of the budget is allocated towards Arduinos, a battery, and a ruggedized case to keep everything protected from the weather.

# 6. Conclusion

We are confident that we can get a successful prototype built given the current timeline and scope of the project. Our research has shown us that our plans our feasible, so the only thing left to do is start the implementation. With the completion of the project, Dr. Apple will not only be receiving more data, but she can do so from the comfort of her home or office.

# 7. References and Appendix

## 7.1 References

[1] “HOBO UTBI-001 TidbiT v2 Water Temperature Data Logger,” GlobalTsetSupply.com. [Online]. Available: https://www.globaltestsupply.com/product/hobo-data-loggers-utbi-001-tidbit-v2-water-temperature-data-logger. [Accessed: 26-Oct-2019].

[2] Adafruit Industries, “SHT-10 Mesh-protected Weather-proof Temperature/Humidity Sensor,” adafruit industries blog RSS. [Online]. Available: https://www.adafruit.com/product/1298. [Accessed: 26-Oct-2019].

[3] Arduino, Arduino Uno Rev3. [Online]. Available: https://store.arduino.cc/usa/arduino-uno-rev3. [Accessed: 26-Oct-2019].

[4] RockSeven, “RockBLOCK Mk2 - Iridium SatComm Module,” SparkFun Electronics. [Online]. Available: https://www.sparkfun.com/products/13745. [Accessed: 26-Oct-2019].

[5] Sleepy Pi, “Sleepy Pi FAQ,” Spell Foundry. [Online]. Available: https://spellfoundry.com/sleepy-pi/sleepy-pi-faq/#Does\_The\_Raspberry\_Pi\_Have\_A\_Sleep\_Mode. [Accessed: 26-Oct-2019].

[6] TODO: Add for wireless sensors

[7] Iridium, “Iridium Edge,” Iridium Satellite Communications, 19-Feb-2019. [Online]. Available: https://www.iridium.com/products/iridium-edge/. [Accessed: 26-Oct-2019].

[8] Globalstar, “STX3,” Globalstar. [Online]. Available: https://www.globalstar.com/en-us/products/embedded-solutions/stx3. [Accessed: 26-Oct-2019].

[9] jonoxer, “practicalarduino/SHT1x,” GitHub. [Online]. Available: https://github.com/practicalarduino/SHT1x/blob/master/examples/ReadSHT1xValues/ReadSHT1xValues.pde. [Accessed: 26-Oct-2019].

[10] RockSeven, “What's RockBLOCK?,” What's RockBLOCK? [Online]. Available: https://docs.rockblock.rock7.com/docs. [Accessed: 26-Oct-2019].

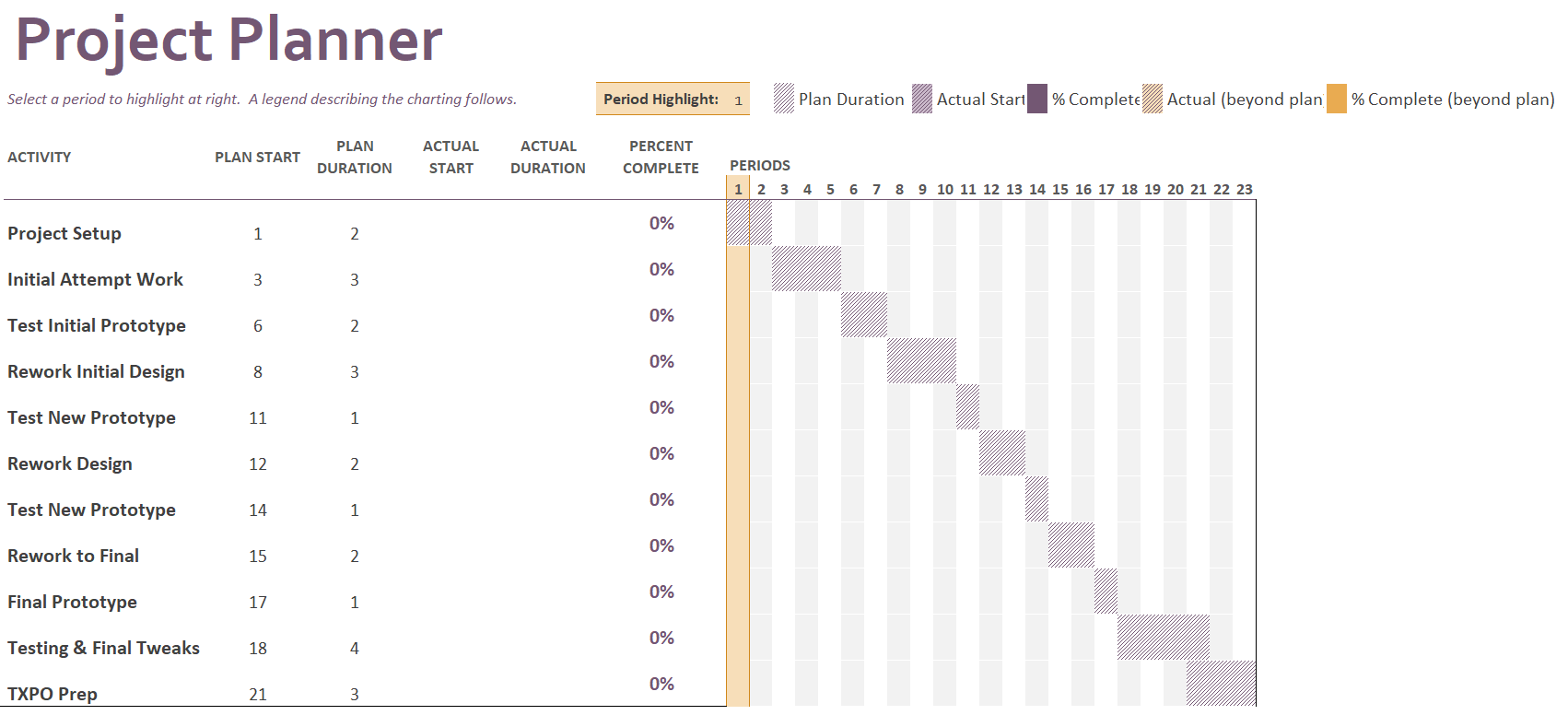
[11] A. Kurk, “Tutorial:A guide to putting your Arduino to sleep,” The Arduino Maker Man, 26-Jan-2018. [Online]. Available: https://thekurks.net/blog/2018/1/24/guide-to-arduino-sleep-mode. [Accessed: 26-Oct-2019].

## 7.2 Appendix

### A. Bill of Materials

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item # | Description | Qty | Unit Price | Price |
| 1 | SHT-10 Temp/Moisture Sensor | 4 | $ 49.95 | $ 199.80 |
| 2 | RockBLOCK Satellite Modem | 1 | $ 273.95 | $ 273.95 |
| 3 | Arduino Uno Rev 3 | 2 | $ 26.20 | $ 52.40 |
| 4 | RockBLOCK Monthly Line Rental | 7 | $ 15.47 | $ 108.29 |
| 5 | RockBLOCK Message Credits | 500 | $ 0.13 | $ 65.00 |
| 6 | Battery | 2 | $ 25.00 | $ 50.00 |
| 7 | Ruggedized Case | 1 | $ 30.00 | $ 30.00 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  | **TOTAL** | $ 779.44 |

### E. Schedule Gantt Chart



### F. Project Overview Block Chart

