Damian Avery

(W963h945)

Smart Park Project Proposal:

Gateway design and signal boosting

To improve our devices signal I must first have a node and a gateway. During last semester we had made strides to create the lora infrastructure needed to send data from the device to our Smart pack app. This semester I have a working node but no longer have a working gateway (originally provided by Go Create). The solution is to configure our own gateway and lora infrastructure.

Design:

I am currently using a RAK 2245 and a Ras Pi 4 to fulfill my gateway requirements. I will also be purchasing a small NEMA rated enclosure to house the components and will more than likely use an Omnidirectional antenna made of fiberglass to finish the build. Below is the key info of the current gateway setup.

Key Info:

Gateway Channel – US915 **16-23** +66 (frequency hop channels)

Gateway ID- e45f01fffe17080d

Operating system- Rasberry Pi 4 OS-Full gateway operating system (provided by Chirpstack)

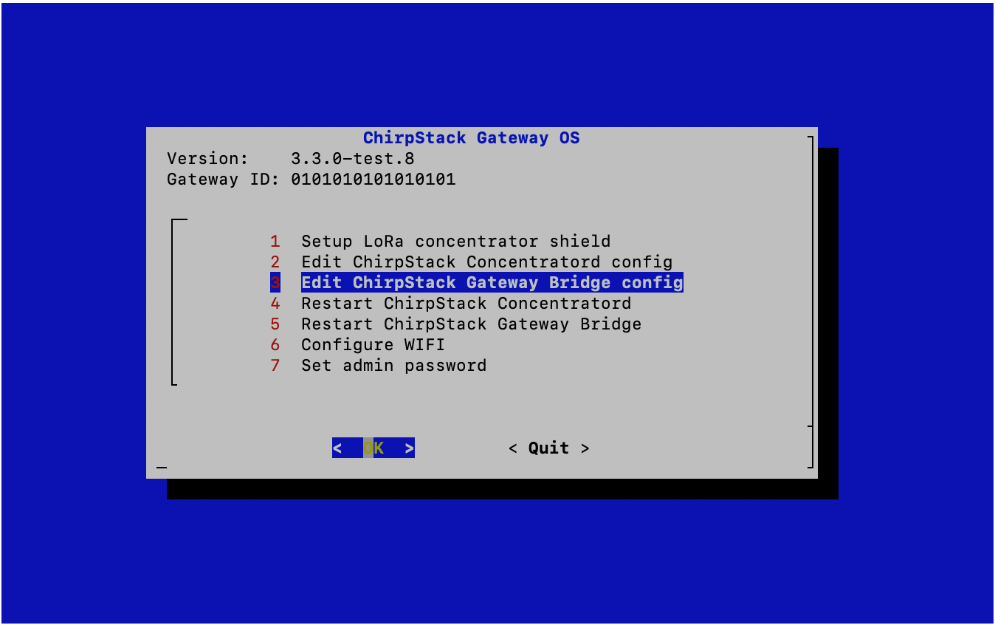
Application Server- GCP virtual machine N2 8GB (Ice Lake)

Network Server- GCP virtual machine N2 8GB (Ice Lake)

Broker- MQTT

Application- smart-park-342405

<https://artifacts.chirpstack.io/downloads/chirpstac-gateway-os/rasberrypi/rasberrypi4/3.5.1/>



To continue the design, I am currently using a virtual machine from GCP (Google Cloud Platform) to act as my Chirpstack network and application server. Below is a screenshot of the current Smart Park dashboard used to house the additional servers.

Graphical user interface, application

Description automatically generated

Testing & Analysis:

Once the gateway infrastructure is completely configured, I can begin to test signal integrity of our current node. I have implemented a standard test plan I will be using for all tests to collect data during testing. I will first begin by sending several uplinks with default settings from variable distances to include behind buildings and other obstructions. I will record each successful and unsuccessful uplink and determine the reliability of our wireless communication. I will then test with an omnidirectional, monodirectional, and bidirectional antenna of different polarities at variable distances. Lastly, I will try to increase the signal power to 20 dbm using the optional PA\_BOOST setting that must be implemented in the code of the Arduino LoRa module. All these results will be recorded using the table in the test plan provided at the bottom of this proposal.

Calculations:

I will use the results collected above to calculate the reliability of our node in different variations to determine the best set up going forward. I will also calculate and record the amount of power usage required to boost the signal. I will determine if boosting the signal is feasible for multiple uplinks a day and a battery life that’ll last approximately 6 years. Below is the equation used to extract the power usage of a 20 dbm signal.



Current progress:

I have been solely focused on deploying our gateway before being able to perform the above tests. The gateway must be configured first to be able to test signal integrity. This was an unexpected hiccup in our project efforts that will soon be solved. See below for full test plan.

Test Plan: Signal Integrity

Date: 02/12/2022

Hypothesis:

The signal integrity of the MKR WAN 1310 has been inadequate. The problem may lie in the capability of the microcontroller. The antenna may not be capable of reliably connecting due to low power or the manufacturer intended for shorter distances. Obstructions such as buildings could also be a culprit as well as the mode set on the component. I must determine if the current hardware is suitable for continued use of our application. We need thousands of nodes to be able to reliably send signals to a gateway.

Plan:

First step is to locate literature on the MKR WAN 1310 and thoroughly look up the signal distance of the antenna. Depending on the results of this search the next step will be to connect a module to the gateway and send signals from varying distances. This will require regaining access to the Chirpstack server to observe data received by the gateway. A large sample size of tests will be performed to provide a reliable statistic of the results collected. Lasty the Module will be tested using the high-power mode and results will be compared.

Results:

Step 1 Research (literature)

7.2 On page 6 of the Sub-G Module Data Sheet discusses the LoRa transceiver Specifications. Under the LoRa Transmitter Specification table below it is observed that different modes can be set to achieve a longer range while consuming more power. Using the PA\_BOOST pin we can reach a 20 dBm signal (decibels relative to one milliwatt). The power consumption can be calculated using the equation below.

Table

Description automatically generated



20 decibels relative to milliwatt consumes 100 milliwatts of power. I was not able to determine the capable range of the transmitter. I did however read on a forum (not the reliability I was hoping for) that the range of the microcontroller could reach up to 6 miles. The entire LoRa network and wireless communication type is meant for Long Range data transmission, so I do not believe that the module is not intended for long range connectivity.

Step 2 Uplink Test (Will not be completed until 02/27/22)

I will now prepare for the gateway connection to test the uplinks and collect the data using the table below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Signal Integrity Test | | | | | | | |
|  |  | No PA\_BOOST | | PA\_BOOST | | No Total | PA Total |
| # | Distance | Signal | Uplink | Signal | Uplink |  |  |
| 1 |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |  |
| 11 |  |  |  |  |  |  |  |
| 12 |  |  |  |  |  |  |  |
| 13 |  |  |  |  |  |  |  |
| 14 |  |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |  |

[dBm - Wikipedia](https://en.wikipedia.org/wiki/DBm)

[Type ABZ | CMWX1ZZABZ | Datasheet | LoRa Module | Murata Manufacturing (arduino.cc)](https://content.arduino.cc/assets/mkrwan1310-murata_lora_module-type_abz.pdf?_gl=1*d51rbg*_ga*MjA5ODU3NDE1My4xNjQ0NzE5MTQw*_ga_NEXN8H46L5*MTY0NDcxOTEzOS4xLjEuMTY0NDcxOTE0MC4w)