A Wireless Sensor Network to Harvest Spatio-Temporal Solar Insolation Data in a Close Proximity

Son Pham, and Saeed D. Manshadi

***Abstract*— Solar energy involves in our daily life especially when renewable energy is being utilized broadly. In this educational project, we create a simple solar radiation tracker using LoRaWAN with ATmega328p microcontroller. This tracker is powered by a battery and solar panel.**

I. INTRODUCTION

As fossil resources reduce, we rely more on renewable energies from the sun, wind, and waves. Humans have been using solar energy many years ago. In order to harvest the most energy from the sun, we make a solar tracker and put it in different spots to find the best location for solar panels. Another application of solar trackers is to track and predict the upcoming weather condition. For example, assume we have a solar farm with tilted panels and trackers that are a mile or two away. By sensing the condition miles away, we can control the panels in the direction that has the most light. According to a paper from Bolívar, and Silva [1], we can use Raspberry Pi as a UDP server and an optical sensor to sense solar irradiance. That's a closed and low-cost project, but the disadvantage is that it's not modular. What we've learned from them is that instead of using an end-product pyranometer we can use an optical sensor to lower the cost of each node. Our final decision is to use an ATmega328p because it has well-supported libraries and non-expensive modular boards to make a quick prototype.

II. DESIGN

A. Design Goal

Our minimum requirement is to create nodes that are able to read voltage from a self-powered pyranometer sensor and send those data to a server. Also, the node has to be able to run continuously for a year with a battery and solar panel. For additional requirements, the nodes need to read voltage, current from the solar panel, and voltage from the battery. The data will be helpful for predicting solar irradiance to improve the electric grid system.

B. Hardware

List of items that we used:

- Voltaic Solar panel: 2 watt
- Battery: 18650
- Current sensor: INA219
- Battery charger: TP4056
- Boost Converter: MT3608
- Transceiver: LoRa Adafruit RFM95W
- Self-Powered Pyranometer: Apogee SP-110-SS
- ATmega328P-PU Microcontroller

There are a few options that we can use for the network such as GSM, SigFox, and LoRa. GSM has wide coverage; however, we have to subscribe to carriers in order to use it. SigFox leans toward business implementation. Also, Our nodes are located in a rural area where SigFox may not reach there yet. Finally, we choose LoRa as our communication technology.

We are using ATmega328p because there are many libraries available for us to implement our ideas and demonstrate proof-of-concept. After the device is up and running we will improve by removing unnecessary code from the library.

We reduce the complexity by using a self-powered solar pyranometer and an Atmega328p as controller. Additionally, we decide to work with modules to save time and experience new concepts. INA219 is used to measure current and voltage output from the solar panel. To charge the battery we use a TP4056 module with constant current/voltage charging mode. In order to keep the sensor running continuously, we supply it with a 3000mA battery. Because ATmega328p's input voltage is 5V so we also use a boost converter to boost up the voltage from the battery. In this proof of concept, we remove a SD card module and a real-time clock module to reserve power. Our goal of this project is to transmit a signal at 1 miles range so we decide to use the LoRa module at 915 MHz which is compliant with the FCC for ISM band. We use a dual-channel LoRa gateway LG02 because a multi-channel gateway is more expensive and overqualified for this project. Because of the dual channel, LG02 does not support data adaptive rate which is a good feature in LoRaWAN to adjust the spreading factor depending on the signal strength. Due to that reason, we only use one channel at a specific frequency for all of our nodes. We use LoRa Calculator from Semtech to calculate time on air and set the time delay for each node to avoid overlapping. According to the FCC, the maximum power for LoRa is 20 dBm. So we set the maximum power for each node to 17 dBm for transmitting. To extend the battery's life, first, we implement sleep mode in which the node will sleep for a minute, then wake up, read the sensor, and send data to the gateway. We also set a threshold value from the pyranometer so that a node will sleep for a longer time at night, but it also reports data every hour.

The authors are with the Department of Electrical and Computer Engineering at Sand Diego State University, San Diego, CA 92182. spham@sdsu.edu; smanshadi@sdsu.edu

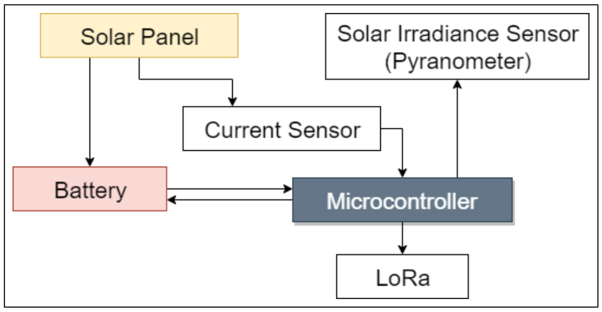

Fig. 1. Setup for the nodes

Fig. 2. Gateway(top left), Opened and Closed Enclosing

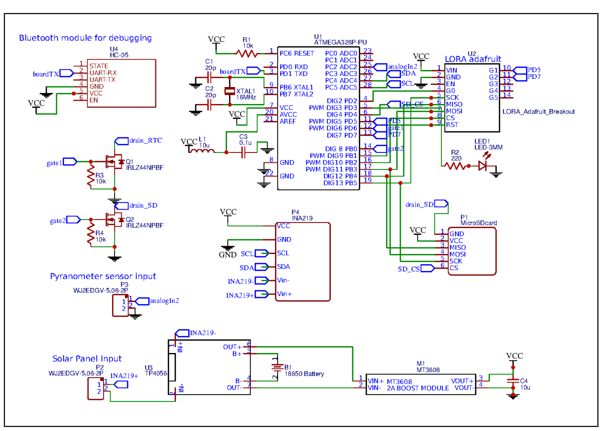

Fig. 3. Schematic

C. Interface

We use the LoRa Serialization library to encode the message. It allows us to transmit up to 32 bytes of payload and also includes decimal numbers. After trying to build our own web app with React, we found the AllThingsTalk web interface which can link to Thethingsnetwork and store our data for up to 30 days. AllThingsTalk allows us to add different kinds of charts in real-time.

III. IMPLEMENTATION

TABLE I
LORA PACKAGE

LoRa package	
Spreading factor(SF)	10
Bandwidth	125 kHz
Code rate	4/5
Payload length	8 bytes
Preamble length	8 symbols

After testing, the node is able to run 24/7 with sleep mode and the range is about 1 mile. We also switch the default antenna with a higher gain antenna to improve the range.

We don't spend too much time on the enclosure because we can use waterproof junction boxes in any hardware store. Then, we apply a sealant to prevent water from leaking through the holes for wires. One of the concerns that we have is the temperature at Imperial Valley which can affect the battery's life and circuit.

A. Problems and Solution

ADC from the Atmega328p is not stable, so we take hundreds of samples and average them out. The sensor's output voltage is more accurate when it's greater than 5.0mA. Also, the percent error compare to the workbench multi-meter is less than 5

IV. CONCLUSIONS

We have not used all the features of LoRaWan, especially sending information to a node from a gateway to control a node. There are many things to improve such as replacing boost converter with low power linear voltage converter. The other improvement we can make is to add additional male headers so we can flash a new code without taking the microcontroller out.

APPENDIX

LoRa: Long Range
LPWAN: Low Power Wide Area Network
UDP: User Datagram Protocol, a simple transmission protocol without error checking.

ACKNOWLEDGMENT

The library that we used: LMIC from IBM, Wire.h, LowPower, LoRaMessage. The PCB board is designed by Marhlo Aposta.

REFERENCES

[1] L. E. P. Bolívar and G. A. da Silva, "Solar radiation monitoring using electronic embedded system raspberry pi database connection mysql, ubidots and tcs-230 sensor," in *2015 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*. IEEE, 2015, pp. 473–479.