Agricultural Radio

High Level Design
13 December 2021

Leo Fangmeyer
Peter Harkins
Ricky Pantin
Mike Pitz

1 Introduction

In this project we are trying to develop an efficient long-range and low-power information processing and control system for agricultural applications. Our main aim will be to develop a proof of concept for a limited number of features, with the intent to add an arbitrary number of features in the future.

2 Problem Statement and Proposed Solution

The ability to send radio messages over longer distances (up to 10km) in such a way that uses very low power is something that is under-utilized. Therefore, smaller devices that shouldn't need to use large amounts of power may die sooner than they should. We want to create a network of devices that can efficiently carry out relatively simple tasks and processing with minimum power use in order to maximize longevity and cost-efficiency.

This project consists of a control system with two types of modules. The first type module, Module 0, is the "control center" that communicates with the sensor(s) out in the field, which belong to the Module 1 category. The Module 1 devices will contain (I) a means of monitoring the environment and (II) a means of communicating with the control center device when necessary. I. In order to monitor the environment, Module 1 may have any combination of thermometers, barometers, pH meters, moisture meters, and photoresistors. Ideally, each Module 1 "field device" would be equipped with as many sensors as possible, allowing for as complete information as possible, though our device will initially only include one or two sensors to establish proof of concept. II. The field device microprocessor can then periodically check that the environmental conditions are appropriate (as well as take interrupt-style inputs for serious environmental deficiencies), and notify the control center Module 0 device when conditions are unsatisfactory. With the communication from the field device, the control center 9 (Module 0) can then activate some other system or device to treat the unsatisfactory environmental condition, such as turning on sprinklers when moisture level is low. For our purposes, however, we do not plan on designing and building a third device to carry out these tasks, but rather plan instead on simply sending a confirmation message to the field device that the control module has received Module 1's message and actuated the proper remedy. We may simulate and display this process by Module 1 turning on an LED to indicate poor conditions, and then turning a different color once Module 0 has indicated reception and initialization of treatment, and then another color (and then off) once the proper environmental conditions are restored

3 System Requirements

Give the overall system requirements. Meeting these requirements will make your system solve the problem you are trying to solve.

- *What is the nature of the required embedded intelligence?*
- *How is the device powered? If it runs on batteries, what kinds of batteries are used, how long should the system be able to run on the batteries, etc?*

- *There are lots of requirements related to wireless interfaces. How many devices need to be supported? What range is required?*
- *What are the user interfaces?*
- *How is the system installed and used?*
- *If you project involves voltages and or currents that may be dangerous, what are safety requirements associated with your system.*
- *What are the mechanical requirements, such as weight, size, etc.*

NOTE: This is not intended as a question and answer section. These questions are just there as a guide.

The system needs a network of devices that can communicate via LoRa. The field device (Module 1) should operate with minimal power consumption to allow for long-term use before replacing the power supply (or entire device). The field device should be able to take in raw sensor data and transmit that data to the base/control device (Module 0). The base device should process this data and determine whether it requires some sort of control based on a set of field specifications for the farm. Based on this logic, it can send a signal to the base device that can actuate some process, including (a command signal), but not limited to turning sprinklers on/off, turning lights on/off, opening or closing gates, dispensing feed for animals, and more. There is also the possibility that the control device will actuate field devices independent of sensor values, such as scheduled feeding times, and also that a field device will sense data that may not be directly acted on all the time. The field device does not need to do any logic processing, but will be limited to sensing and transmitting data, as well as enacting the actuation when commanded by the control device to some relay device. Some ideas are to use the STM32WLE5Cx, ATtiny426 8-bit MU, or the ATtiny1626/1627) for the field device and the ESP-32 for the control module, which will provide WiFi and Bluetooth capabilities. The STM32WLE5Cx seems like a good choice for our remote devices, since we will need to integrate LoRa capabilities in one way or another, and this microcontroller could solve our problem by integrating LoRa and the microcontroller into one device. However, we may also opt to use separate microcontroller and LoRa transceivers, like the ATtiny family alongside something like the COM-13909 LoRa transceiver. This would likely save a few dollars, though will likely make design slightly more complicated, and could even sacrifice some black-box integrated functionality offered by the integrated MCU-LoRa device.

The base device will need to communicate with multiple devices in the field; for our purposes, we will make 2 field devices to ensure that the base device can flexibly and intelligently communicate with more than just one device. They should be able to communicate over ranges over 2 miles.

The remote device will be powered with lithium-ion cells, and should last for at least a year or two before needing any replacement or troubleshooting. As such, we need it to be very low power, which means actuation of relays may be challenging. If this becomes too difficult, we may need to explore other means of carrying out actuation

The primary user interfaces will be the html page that will display data recorded and will log the various actuations. In order to actuate from the base station manually, the html page will have a GUI of some sort.

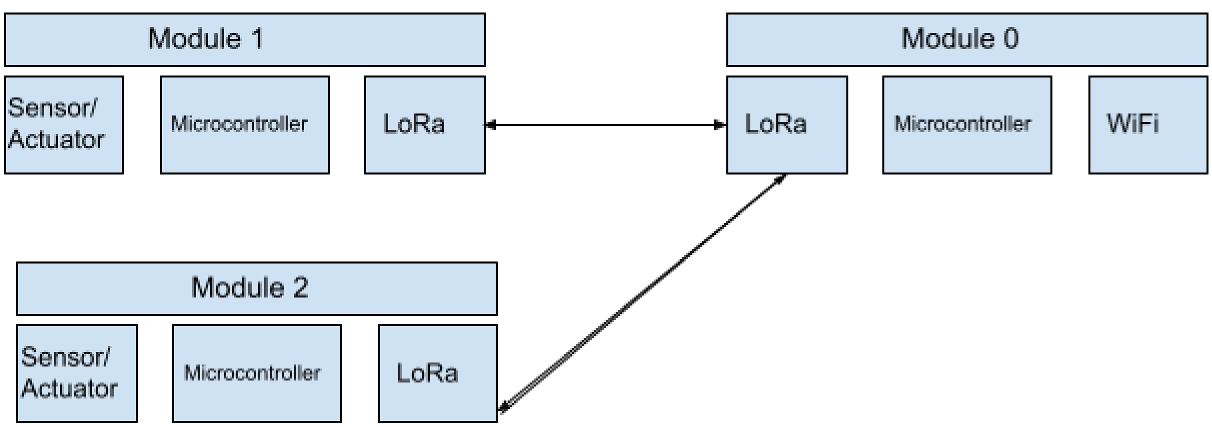
The installation is relatively simple. The field devices can be placed in predetermined locations that the property owner wishes to collect data on. The circuit board will be in some sort of housing that will protect it from the elements and possibly help it remain in one place. The base module can just be powered by some sort of USB, likely from wall voltage.

4 System Block Diagram

4.1 Overall System:

I find it easiest to proceed with making design decisions by first breaking the system down into subsystems. Note this can be an iterative process. For example, I might choose to group the user interface (input and output) into a single subsystem. While this might make sense, if there are multiple outputs and inputs (such as an on-device display and a smart phone interface display) it might make sense to have separate blocks for different parts of the user interface.

The system requirements must be met by the system as a whole, and those requirements naturally migrate to subsystems that comprise the system. Depending on the system decomposition, requirements may be met entirely by a subsystem, or through the combined actions of multiple subsystems.



4.2 Subsystem and Interface Requirements:

The requirements of each subsystem or major interface are described here by subsystem. These lower level requirements support the overall system requirements. Note that major interfaces (such as a wireless interface) should be described like any *other subsystem. Don't forget that there will be software as well as hardware in many of the subsystems, and that software will have requirements.*

- Base Module
    - Microcontroller
        - The microcontroller for the base module must have wifi capabilities that can be utilized to store data that is received from the remote modules
        - Will need to have LoRa capabilities.

    - Power Supply
        - The base module's power supply must be a permanent connection so as to be able to constantly be able to receive data from remote modules
    - Data Logging
        - Base module must be able to receive data by means of a LoRa transceiver, as well as transmit this data to the user's desired method of data storage
    - Control Logic
        - The base module must be able to interpret/process incoming data and send a signal to remote actuators if certain conditions are met
- Remote Modules
    - Microcontroller
        - The microcontroller on the base module need only have enough processing power to be the middleman between the sensors/actuators and the base module.
    - Power Supply
        - The power supply must be able to power the remote module (along with its ancillary features) for at least one season/year. It must also be small enough to fit into a weatherproof enclosure
    - Sensors
        - Sensors must be able to accurately detect temperature and moisture levels and efficiently communicate this data to the microcontroller
    - Actuation
        - The remote modules must be able to perform simple actuation, as controlled by the base
    - Weatherproof enclosure
        - The remote modules must be placed in a weatherproof enclosure so as to not cease operation during adverse conditions
- LoRa Interface
    - The LoRa radio interface must be able to effect long-range wireless communication of sensor data that is sufficiently detailed (number of bits). It must also accomplish this with minimal power consumption in order to not drain the remote module's battery
- Data Storage/Logging

- Receive information from the base module over WiFi
- Store it somewhere with a more accessible interface

4.3 Future Enhancement Requirements

There may be a number of features that aren't going to be part of the initial release of your product, but that you would like to add in the future. These are listed here so that the design does not preclude adding these features.

- Additional sensors (light, pH, motion, etc.)
- Solar power (rather than battery)
- Actuators and Relays (see questions)
- Streamlined user interface- mobile application
- More frequent measurements and/or more locations

5 High Level Design Decisions

Broken down by subsystem and major interface, this section presents your high level design of each subsystem or interface. ***Your design decisions should be guided by choosing options which best support your system requirements.***

For each subsystem or major interface, you should describe the function or interface and the devices that will be used to realize the functions performed by the subsystem. The decision level is not to the level of a complete schematic, but it is necessary to identify the major components that will be used because those choices affect other design decisions.

If a subsystem contains embedded intelligence, it is not necessary to specify the specific microcontroller that you will be using. The requirements listed earlier should allow you to specify a class of microcontrollers (based again on requirements like cost, power (electrical), power (processing power), I/O and interface requirements, etc.)

Keep in mind related issues, like how each subsystem is going to be powered, how clocked devices will get required clocks, etc.

- Base Module
    - Microcontroller
        - The microcontroller on the base module will most likely be the ESP32 (or a very similar chip) as this has decent processing power, as well as both WiFi and bluetooth capabilities that will facilitate communication of the data to an HTML page
    - Power Supply
        - A practical permanent power supply for the base module is some type of USB connection coming from a wall receptacle
    - Data Logging

        - The data from remote modules will be received via a LoRa module. This data will be communicated to the microcontroller via a serial interface (most likely SPI). This data will then be sent over WiFi to an HTML page for storage and for easy interfacing with the user
        - Potentially use SD card in order to store data without power.
    - Control Logic
        - The base module will also be able to automatically react to conditions based on incoming data. We will implement interrupts that, when triggered, cause the base module to send out actuation instructions to one of the remote modules.
- Remote Module
    - Microcontroller
        - The microcontroller on the remote modules will have less processing power than that on the base module. It will utilize I2C interfacing with the various sensors, and some other serial interface to communicate with the LoRa module. It should definitely be as low-power as possible
        - We plan on using either:
            - STM32WLE5JB (LoRa is integrated)
            - ATtiny426 8-bit MU or the ATtiny1626/1627 with a LoRa transceiver (COM-13909)
    - Power Supply
        - The power supply for the remote modules will be a long-lasting battery. The battery will also be able to be easily replaced and rechargeable
    - Sensors
        - We will include moisture and temperature sensors that have I2C interfaces in order to communicate with the microcontroller
    - Actuation
        - We will use LEDs to indicate specific actuation commands that are being sent to the field devices and properly enacted.
        - Another possibility is to include relays so that the microcontroller can turn additional functionalities on or off. If this is implemented, the microcontroller would have to switch on a transistor that then turns on a relay.
        - We also hope to have manual auction functionality by using a GUI through the html page that would then through WiFi communicate with the base device to then send a LoRa signal to actuate at the base.
    - Weather-proof enclosure
        - The remote modules will be enclosed in a capsule similar to those used for outdoor electrical receptacles
- LoRa Interface
    - Depending on what microcontroller we will either:

        - Implement and integrate external LoRa modules onto our PCBs to facilitate the RF communication, such as using a LoRa transceiver (COM-13909).
        - Have LoRa capabilities already integrated into the microcontroller (STM32WLE5JB). Would need to configure the LoRa module in the microcontroller.
- Data Storage/Logging
    - We will design an HTML page to store the sensor data, as well as to allow the user to easily see trends and relationships in the data
    - This will also include a GUI to allow for manual actuation

6 Open Questions

In most of these projects, there will be a number of pieces of the design that are not obvious how they could be realized. In this section, you should list the things that you are not sure how to do, or not sure that what you are planning will work. Some of these I have identified and have asked you to demonstrate this semester. There may be others that you are unsure of. All of the unknowns will need to be addressed before you can commit to a low level design.

It will not be unusual that I will have boards and devices that will allow you to do experimentation without significant cost or development work. If there are components that are not easy to prototype, you need to identify them

- Implementing weatherproof enclosure while also allowing sensors to operate
- Do we need to rigorously implement the activation capabilities of the field devices, rather than just using LEDs? Do we need to create functionality to allow for activation of a relay?
    - We are concerned with the challenge of generating enough current from the microcontroller to work an actual relay that activates a desired system in the farm (sprinkler system).
    - However, there is definitely the possibility of using the microcontroller as a control signal to close a switch/transistor that would allow for a separate power source, such as the power source for the field device, to supply the current to the relay.
    - If we were to use a transistors that turn on a relay we would need to physically connect the microcontroller to the transistor; so the communication between the field device and the relay would not be wireless.

7 Major Component Costs

Specify the costs of your major system components. This should allow you to have a rough cost estimate of your system. This should include any specialized equipment necessary to demonstrate your project.

PCBs: ~$50 each

Remote Module Possibilities:

1a. AtTiny + CM-13909: .74 + 5.95 = $6.69
1b. STM32WLE5JC: $9.90

8 Conclusions

Our project will implement a network of remote modules for sensing and actuation, along with a base module that will accomplish data storage and manipulation. These modules will communicate via the RF protocol LoRa, which will be implemented with external modules on PCBs. This system will allow for accurate tracking of agricultural data over extended periods of time, and it should be straightforward to install for amateur or smallholding farmers.