Project Elective Report

Sai Manish Sasanapuri
Mentor : Prof Madhav Rao

April 30, 2021

Abstract

This project elective aimed to use IoT-based approaches that make farming more controlled and accurate. Throughout this semester I learned different concepts. The important things were to collect different sensors data, most importantly different ways to communicate between Arduino, laptop and mobile. Additionally, tried to efficiently compress the data(pack the data) to reduce transmission power overhead and decompressing the data at the receiver side. Explored different sleep mode of Arduino microcontroller to save power and different ways to store the data.

1 Introduction

In a country like India with a huge population and increase in population, there is a need for an increase in crop production to feed this huge population. But due to climatic changes and lack of rains, drop in groundwater levels, and still following techniques which were evolved hundreds of years ago and doesn't take care of the conservation of resources, the crop production is not up to the mark.
So we can use technology as a bridge to effectively use water and other resources to increase crop production. In the past few years, we could see that many lands came under un-irrigated land due lack of knowledge on how to use resources efficiently. The project aimed to develop a low-cost and low powered system where the farmer can monitor their crop and take actions accordingly.
The system mainly consists of two subsystems,
1) Parent subsystem and
2) Child subsystem.

2 Configuration of system

1. **Parent device subsystem :** This subsystem receives the data sent from child devices, process them and then upload them to the cloud server.
2. **Child device subsystem :** This subsystem is placed in the field for data collection, like air temperature, air humidity, soil temperature and soil moisture.

3. **Repeater subsystem :** This subsystem is for prolonged communication over distances. i.e it is not possible to communicate long-distance using short-range wireless communication devices, so for this, we use a repeater subsystem.

3 Components & Analysis

3.1 Arduino Microcontroller

An Arduino Microcontroller is a multipurpose tool that can be used for many electronics projects as it provides many functions to a particular circuit. For our project, we used to collect sensor data.

Figure 1: Arduino microcontroller

3.2 DHT 22

The DHT22 sensor is a basic, low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and spits out a digital signal on the data pin.

Figure 2: DHT 22

3.3 Capacitive Soil Moisture Sensor

Soil Moisture sensor measures volumetric water content in the soil. This Capacitive soil moisture sensor consists of a single probe contains separate VCC and ground copper paths covered by solder mask act as a flat capacitor. This layer will prevent the direct passing of electric current into the soil. The sensor measures the soil moisture levels by capacitive sensing, Depending upon the water content of the soil, the dielectric of the soil changes. The changing of the capacitance which is corresponding to the soil moisture content.

Figure 3: Capacitive Soil Moisture Sensor

3.4 Soil Temperature sensor (DS18B20)

The temperature sensor measuring the soil temperature should be waterproof and robust enough to corrosion. The temperature sensor (DS18B20) would match all these conditions. It exists in two forms. One is in IC and the other is in a probe like structure. We are going to use the DS18B20 probe type.

Figure 4: Soil Temperature sensor (DS18B20)

3.5 DSRTC 3231 module

The DS3231 RTC module Precise Real-Time Clock Module is a low-cost, extremely accurate I^2C real-time clock (RTC) with an integrated temperature-compensated crystal oscillator (TCXO) and crystal. The device incorporates a battery input and maintains accurate timekeeping when the main power to the device is interrupted. In this project it sends interrupt signal to wake up arduino from sleep mode at regular intervals of time.

Figure 5: RtC 3231 timer

3.6 HC-05 Bluetooth module

HC-05 Bluetooth Module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Its communication is via serial communication which makes an easy way to interface with controller or PC. The HC-05 Bluetooth module provides switching mode between master and slave mode which means it can be used either for receiving or transmitting data.

Figure 6: HC-05 Bluetooth module

3.7 Resistors

Resistors are a chief electrical component in circuits - they are used to reduce current flow and at the same time maintain low voltage levels to prevent short circuits and provide the required current flow in simple circuits. In our project we have used these resistors in several points in the circuit accordingly.

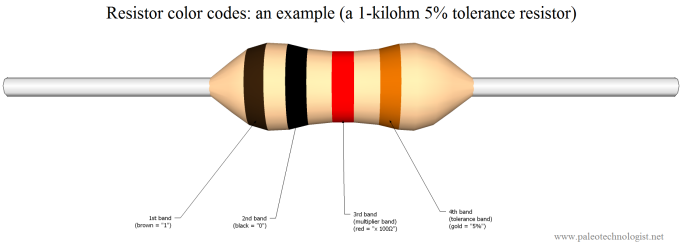

Figure 7: Resistor

4 Project Implementation

The primary objective in this project elective is to transmit the sensor data collected by the child subsystem and transmit using a wireless communication device. Presently I am using the HC-05 Bluetooth module here but we can use Lora devices which is low power long-range communication device.

How it works? Firstly we will connect all the sensors to Arduino to collect field data. We will be using the RTC timer to wake up Arduino by sending an interrupt to Arduino for every specified interval of time. I have tried two ways of storing the data, one is to store the data in the local system as a CSV file or can upload the data to the cloud. In this project, I have used the thingspeak platform as a cloud platform to store the data and plot the received data.

Here python is middleware to save or upload collected sensor data by arduino, using the serial library in python to read the real-time data of all the sensors from the COM port and wrote a python script to write them in a CSV file or upload data to the thingspeak platform.

Assuming that we don't need a precision of 1 or 2 then we can save a lot of bits. If our message packet should have only 32bits, without precision we can pack 2 sets of data(soil temp and atmosphere temp) + 8bits(one data + 2bits can be used for something else) or those 8 bits can used for some encoding scheme. With precision, we can pack only one set of data (soil temp and atmosphere temp).

An efficient way of sending data by reducing bits. If we know the range of

sensor value, we can manipulate them. For example, if we know the temperature range varies in between (-7 to 7 from a certain value) then we can transmit just 4 bits(offset) in place of 6 bits at the receiver side we add this certain value to that offset. By doing this we save 2 bits for each sensor value(without precision). By this method, if the payload size is 32bit we can pack 4 sets of data(soil temp + atmosphere temp). The mean value for which we will be adding the offset is send at the start of the program so that both the receiver and transmitter are in sync with the values.
The Arduino sends the packed data as we are using python as middleware it reads the packed data received through COM port first, and then decode or unpack them and get actual values (by adding mean + offset). The decoded data is processed and then either saved in a CSV file or uploaded to the cloud.

Parameters currently measured :

1. Soil Temperature
2. Soil moisture content
3. Air temperature
4. Air Moisture content/ Humidity

GitHub link for all my codes and workings slides

5 Future Developements

1. We can collect water level data.
2. We use camera for Photo analysis of Nutrient contents in crop
3. We can collect Soil pH level data
4. Can make it semi automated or automated system. In Automated it should compute the sensor values and do the action accordingly. In semi automated the user should check the values and if possible using previous data we can suggest what steps to follow.
5. For further development of the project we can use data-mining and ml algorithms to suggest best crop on soil, weather conditions, and place at which the crop is present.

6 References

Reference for components :

1. Soil Temperature sensor (DS18B20)
2. Capacitive Soil Moisture Sensor
3. Temperature and Humidity sensor (DHT22)
4. RTC timer
5. HC-05 bluetooth module

Other references :

1. Agri-stick
2. Smart way to store data
3. Device Power Management in IoT
4. Arduino- RTC- timer
5. Arduino Data Logger (CSV) with Sensors and Python
6. Soil Moisture Sensor: Innovation For Precision Farming
7. Arduino + Bluetooth + cloud
8. Arduino with HC-05 Bluetooth Module
