A miniaturized radar sensor and GPS system integrated to an Unmanned Aerial Vehicle

Quinn Powell

Xaiobing Zhou

Final Report

Project Objectives

My final goal by the end of this project was to create a miniaturized radar sensor and GPS system that could be integrated to a drone with a lightweight frame to be used as an innovative tool for researchers and other industries.

- Develop an integrated miniaturized radar system including a microcontroller board, a GPS receiver, a radar chipset, a micro SD card for data storage, and ground controlling (WIFI) system.
- Integrate the miniature radar system into a drone with a compact design for easy electronic component and data storage.
- Design a compact box for holding the electronic components and batteries and print it using the 3D printer of the Geophysical Engineering department.
- Field test the radar system after integration to the UAS.

Radar Sensor Design

First, I needed to design the miniaturized radar sensor system that includes a GPS system, a microcontroller, and a data logger. This system would have to withstand factors such as the overall weight the drone can carry and how to power the radar sensor and the GPS. All these pieces would need to be small to be less than or equal to the weight the drone can carry. Another factor I considered was if a component of the system could interface with the Arduino IDE programming software.

Results

Global Positioning System (GPS)

The GPS was tested in the geophysical engineering laboratories and produced accurate results. The adafruit GPS is able to record latitude, longitude, and the altitude above sea level in meters.

Arduino Data Logger and Programs

The Arduino data logging system was tested in the geophysical engineering laboratory and was tested with the GPS and real time clock. This code worked well and was able to provide me reliable and quick GPS and real time clock data.

```
char c = GPS.read();

// if a sentence is received, we can check the checksum, parse it...
if (GPS.newNMEAreceived()) {

  if (!GPS.parse(GPS.lastNMEA()))   // this also sets the newNMEAreceived() flag to false
    return;  // we can fail to parse a sentence in which case we should just wait for another
}

// if millis() or timer wraps around, we'll just reset it
if (timer > millis())  timer = millis();

// approximately every 2 seconds or so, print out the current stats
if (millis() - timer > 0) {
  timer = millis(); // reset the timer
```

This part of the code ensures the GPS receives a fix on its location before it records data.

```
while(GPS.fix){
  File myFile = SD.open("datalog.txt", FILE_WRITE);

  DateTime now = RTC.now();

  myFile.print(now.year(), DEC);
  myFile.print('/');
  myFile.print(now.month(), DEC);
  myFile.print('/');
  myFile.print(now.day(), DEC);
  myFile.print(", ");
  myFile.print(now.hour(), DEC);
  myFile.print(':');
  myFile.print(now.minute(), DEC);
  myFile.print(':');
  myFile.print(now.second(), DEC);
  myFile.print(", ");

  myFile.print(GPS.latitudeDegrees, 7);
  myFile.print(", ");
  myFile.print(GPS.longitudeDegrees, 7);
  myFile.print(", ");
  myFile.print(GPS.altitude, 7);
```

When the GPS finds a fix on its location the GPS data and real time clock data are recorded.

Radar Sensor Code

During this project I was able to write a code to measure the speed of the electromagnetic waves with respect to two-way time travel.

```
#include <SoftwareSerial.h>
SoftwareSerial ops241Serial(12, 11); // RX, TX
void setup() {
Serial.begin(19200);

// set the data rate for the SoftwareSerial port
ops241Serial.begin(19200);
}
void loop() { // run over and over
if (ops241Serial.available()) {
Serial.write(ops241Serial.read());
}
if (Serial.available()) {
ops241Serial.write(Serial.read());
}
}
```

Conclusions

In this project I was able to effectively design and create a light weight radar system that could measure GPS, real time clock, and radar data easily. This system consists of a radar sensor, a GPS, and an Arduino microcontroller / data logger. Future research into this project will involve writing more code to calculate the dielectric constant and electrical conductivity of soils and surfaces. These codes will then be tested with a fully integrated drone system in the field compared alongside geophysical soil moisture sensors.