Air Particulate Sensor POD v2

GOAL:

- Interface Analog gas sensors to detect:
    - Carbon Monoxide
    - Hydrogen Sulfide
    - Methane
    - Propane
- Interface Analog temperature sensor
- Interface Dylos DC1700 air particulate sensor
- Interface MTK3339 GPS sensor
- Interface SD Card
- Interface XB S3B 900MHz radio
- Publish data on a web server using Intel Edison
- Upload data to IBM Bluemix cloud

DELIVERABLES:

The project will consist of a cohesive sensor package that mounts onto a Unmanned Aerial Vehicle (UAV) and transmits the data to a receiving station. Once received the data will be pushed to a web site and stored on an IBM cloud server.

LITERATURE SURVEY:

Air quality is monitored by several agencies in the United States to prevent the deterioration and contamination of the air. This is done by constant sampling and evaluation of air to monitor for pollutants and to keep an accurate database. Many of these pollutants can cause serious health issues and harm the environment. Currently the monitoring of pollutants is performed by a large established network of air monitoring stations. These air monitoring stations have a high construction and maintenance cost. Therefore, a low cost mobile solution needs to be created that could fill in the spatial “gaps” in the fixed monitoring network.

This project developed a mobile aerial platform for air quality monitoring using a small unmanned aerial vehicle (UAV) paired with a low cost air particulate sensor. This is a low cost robotics solution that can eliminate the need for multiple high cost monitoring stations. It can also be used in extremely hazardous environments where a human cannot enter or access.

For example, local governmental agencies in charge of dust abatement could use this product to quantitatively measure dust pollution from construction sites. In Clark County, the county monitors dust pollution by training inspectors to visually estimate the opacity of a dust plume. For an opacity of 20% or greater, the construction firm would be fined $2,000 for the first violation. Recurring violations reach up to $10,000 per day. While the current method of inspection is legal and valid, this product should complement the inspector’s visual training. Conversely, this product could also be used by the construction firm to ensure their dust abatement procedures are effective.

Another implementation for this product would be for the NSTec emergency response team to use this drone in an environmental disaster. The team had an employee from the emergency response team of NSTec inquire about using the drone in the Fukushima Daiichi nuclear disaster. She was very interested in it and in future developments.

The EPA and Los Alamos National Laboratory have also performed projects to monitor air quality using UAV systems. They used fixed wing radio-controlled aircraft and flew them through smoke clouds. They determined that the aircraft travels too fast to take effective samples. Also those drones are not conducive to indoor environments. They published their results at the National Air Quality Conference in Raleigh, North Carolina in 2010 [1]. In late September of 2014, the EPA hosted a conference where researchers used UAVs for air quality monitoring [2]. They noted the legal and technical challenges with using UAVs. They also noted some uses including meteorological measurements, particulate measurements, and gas measurements.

The ultimate goal of the project is to create a sensor module that can be used with any quadcopter and that will be easy to deploy on site. The sensor module has the capability of operating and transmitting data from several sensors as required by the end customer as well as processing, transmitting, and displaying the information in real time. The live information capability is crucial in applications where the drone can be deployed to avoid risk to human life in hazardous environments like fires, where the pollutant or gas concentration can be very high.

COMPONENTS:

The objective of the project is to interface a Dylos DC 1700 sensor, four gas sensors, a temperature sensor, a GPS Module, an SD card, an XBee 900 MHz radio with an ATmega2560 microcontroller.

A. Dylos DC 1700

The first step in the design was to evaluate and understand the existing design and housing for the Dylos sensors PCB board. The Dylos DC1700 was taken apart to observe the features of the housing and how the sensor components were integrated with the PCB. It was noted that 3 holes were needed to accommodate the buttons that activated the Dylos sensor and gave access to its different configurations and operating modes. Also, an air duct had to be constructed to recreate the air flow rate in the handheld Dylos sensor. The air constriction on the Dylos sensor was measured and modeled by adding ink to the constriction walls and placing it on paper. This air constriction creates a steady air flow where the sensor takes the air particulate readings. The air duct and air constriction were modeled in Solidworks and then 3D printed. The Dylos DC 1700 had components that were un-necessary for the operation of the sensor when mounted on the quadcopter. The removed components were the battery, LCD screen, stock fan, RS-232 connector, and battery charging port. The Dylos's PCB board contained a microcontroller that was used to transmit the data to the attached LCD display. This unit was probed with an oscilloscope to locate the TX pin that transmitted the data. This was then connected to the RX pin of the ATMega2560.

B. MTK3339 GPS

The QuaRK is designed to be used for indoor applications; however, the sensor module was designed to be used in both indoor and outdoor applications. Therefore, it was decided to add a GPS module to the sensor package. Several GPS modules were evaluated and compared for cost, performance, and ease of use. The device selected was the MTK3339 GPS module because the Adafruit Ultimate GPS Breakout board using this chip was successfully used in the previous design. The MTK3339 GPS module can track up to 22 satellites on 66 channels with update rates of 1Hz, 5Hz, or 10Hz. According to the product specifications, the GPS module only draws 20mA of current while in active use.

C. Gas and Temperature Sensors

The gas sensors and temperature sensor selected were analog devices. The gas sensors selected for the sensor module were the MQ-2 (CH_4), MQ-5 (C_3H_8), MQ-7 (CO), and MQ-136 (H_2S). These gas sensors are analogue devices and work at 5 volts drawing approximately 125 milliamps of current each. These sensors have a detection range of approximately 200 ppm to 10,000 ppm for all gases exept H_2S. The MQ-136 has a detection range of 10ppm to 110 ppm. These gas sensors were selected due to the limitation of funds and price gap for better quality non-analog gas sensors. The manufacturer of the gas sensors is Hanwei Electronics. The temperature sensor selected was the TMP36. The TMP36 has an operating temperature range of -40 °C to 125 °C with a ±1 °C accuracy. This temperature range is within the expected operating temperature range of the sensor package and quadcopter.

The data sheets for these analog sensors were analyzed and their characteristic functions were calculated from the functions graph. The TMP 36 is a linear function that results in the following equation:

$$x\,mV = x{^\circ}C + 50{^\circ}C \qquad (1)$$

The gas sensors are linear functions that are graphed on logarithmic graphs. The y axis of these graphs is the value of the variable resistance of the device labeled RS/RO. The x axis is the concentration of the gas in parts per million (ppm). In order to calculate the gas concentrations a voltage divider is used with the internal resistance of the device in series with a load resistance. By calculating the value of the output voltage we can solve for the internal RS/RO value and obtain the corresponding gas concentration. The formula for the logarithmic equation is given as:

$$y\ RS/RO = kx^m\ ppm \quad (2)$$

This equation has two parts, k and m. The intersection where xm is equal to 1 is the k value and the slope of the line is the m value where m equals Δ log y/Δ log x. Table 1 shows the calculated equations for each of the gas sensors.

D. Microcontroller

The microcontroller selected for the project was the ATMEGA2560. The previous sensor module used two Arduino Uno's using ATMEGA328P microcontrollers because one microcontroller did not have enough memory to process all the sensors data. The addition of 5 sensors required a more powerful microcontroller that had enough memory to process the sensors data. The ATMEGA2560 has 256kB of memory compare to 32kB for the ATMEGA328P and it comes in a TQFP-100 package. The TQFP-100 package is approximately 12x12 mm. The gas sensors and temperature sensors transmit a voltage between 0-5V. The microcontroller uses the Analog to Digital Converter (ADC) to process the data into gas concentration and temperature readings. The Dylos DC1700, GPS module, and XBee radio used serial communication to receive and send information. The ATMEGA2560 used the Universal Asynchronous Receiver Transmitter protocols. It is important to know that the pins are labeled as transmit (TX) and receive (RX). The assigned RX and TX pins on the microcontroller should be connected to complementary named pins on another device. For example, the TX pin on the microcontroller should be connected to the RX pin on the sensor and vice versa. The data transmission speed should match on all devices for effective communication between devices. A rate of 9600 bits per second (baud) was selected for all devices since it is the most common rate used in serial communication, and also because this provided far more data capacity than needed for the anticipated rates of data communications from the sensor package. It is important to note that the microcontroller can only receive information serially from one device at a time. After the information was processed by the microcontroller it was stored in a SD card and also transmitted to a computer to allow real time monitoring. A generic surface mount SD card reader was used in the PCB design.

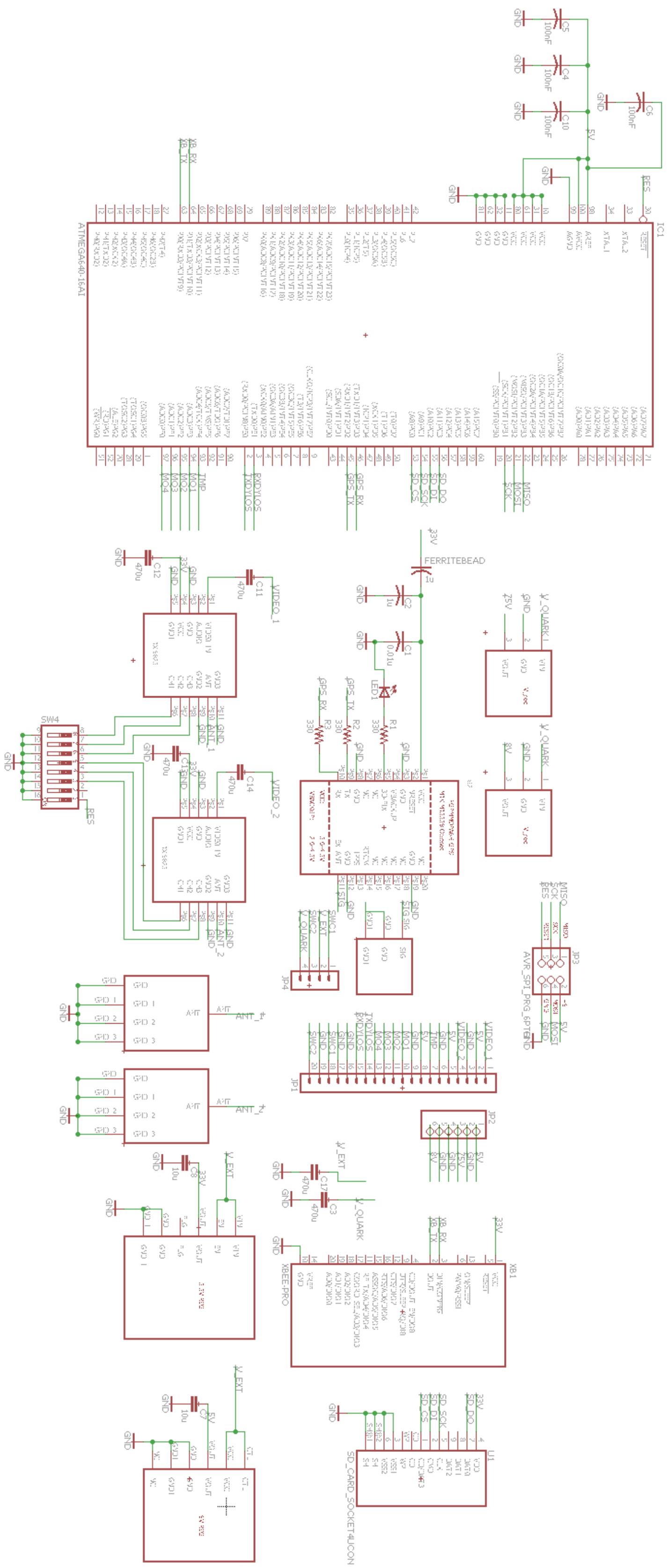
E. Radio Transmitter

The selection of the radio used to transmit the information was very important. The controller for the quadcopter transmits on a 2.4GHz frequency, thus a radio transmitting the sensors data on the same frequency is at risk of experiencing interference. The other options left on the unlicensed frequency band are 5.8GHz and 900MHz. The problem with transmitting the information on 5.8GHz is that the range is less than 300 meters without obstructions, and the video is transmitting at that frequency. The range can be slightly extended but at the expense of using more power. At 900MHz the transmit and receive ranges are greater, and less power is required to transmit the information, but sometimes a higher gain antenna needs to be used. The radio selected to receive and transmit information was the XBee-Pro S3B. The XBee S3B operates at 900MHz and has a maximum range of 6 miles line-of-sight in perfect atmospheric conditions. This radio outranges the controller for the quadcopter. The XBee radio also has an adjusTable power setting for the transmission of data. The maximum power setting consumes 250 milliwatts (mW).

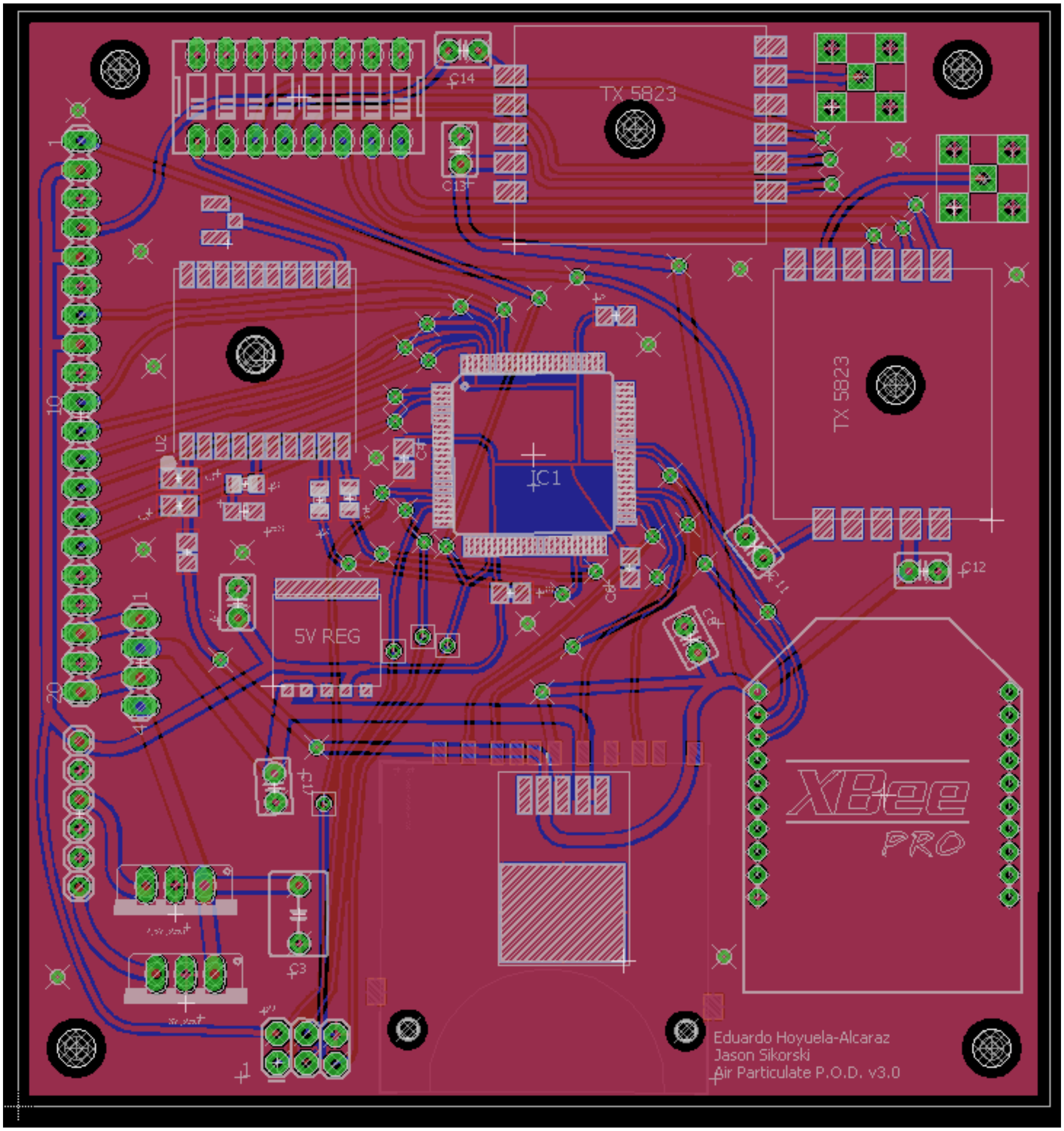
F. External Battery

The electrical load was calculated, and it was decided to use an external battery to power all the electronics requiring 3.3V and 5V. The Dylos DC1700 and 7.5V fan remained operational under the main quadcopter battery. The external battery had to provide a voltage above 7V for the 5.5V and 3.3V voltage regulators to operate, had to be lightweight, and provide enough current to operate the electronics for at least 15 minutes. The battery that met these requirements was the Thunder Power ProLite+Power Series 2-cell 7.4V 730mAh battery. This external battery weighed 39 grams and had dimensions of 12x31x55 mm.

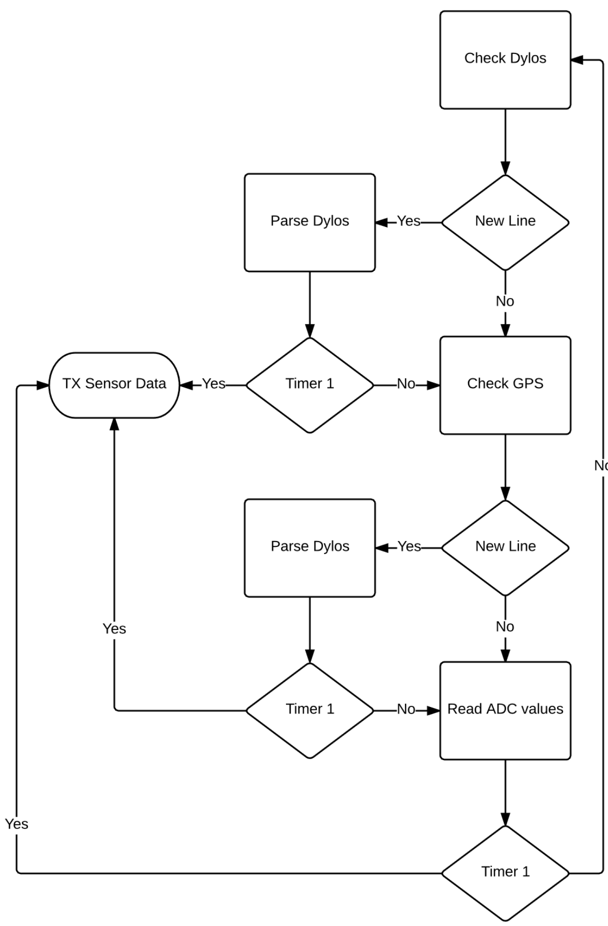
SCHEMATIC:



INITIAL PCB:


TX 5823
TX 5823
IC1
5V REG
XBee
PRO
Eduardo Hoyuela-Alcaraz
Jason Sikorski
Air Particulate P.O.D. v3.0

IMPLEMENTATION:



The code for this project uses interrupts on the UART transmission of the Dylos sensor and GPS sensor. When a character is received it is appended to a string and checked for '\n' character. Once the new line character is received a flag is set noting that the data is ready to be parsed. There is also an interrupt on Timer 1 that sets a transmit flag. When the transmit flag is set the data string is sent via UART to the radio transmitter. Once the Dylos and GPS modules have been checked each of the analog sensors calls a function that returns their value. Once all the values have been returned from the sensors a human readable string is generated and sent to the SD card while a comma delimited string is stored and sent to the radio each time Timer 1 overflows.

CODE:

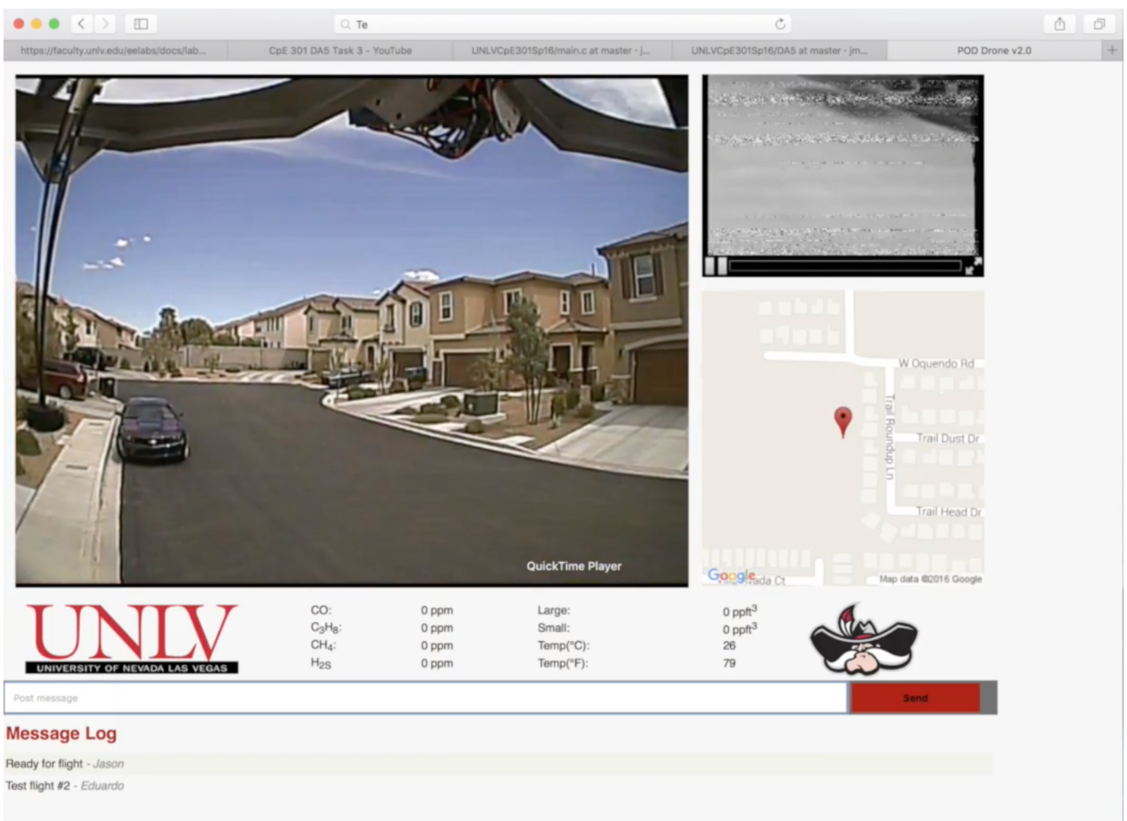
Code is hosted on GitHub and can be retrieved at: https://github.com/jmsikorski/UNLVCpE301Sp16/tree/master/Final%20Project

VIDEO:

Video demonstration is hosted on YouTube and can be retrieved at: https://youtu.be/t8fbz6CZ9Ew

SCREENSHOTS:

IBM Cloud Visualization



Web App Implementation