COMBINING MULTIPLE SENSORS ON ADAFRUIT FEATHERLOGGER

Draft of February 9, 2023 at 18 : 59

BY

ARNAV SHETH

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Bachelors of Science in Electrical and Computer Engineering
in the Undergrad College of the
University of Illinois at Urbana-Champaign, 2025

Champaign, Illinois

Adviser:

Jiang Chongya

ABSTRACT

The main goal of this project is to take frequent measurements from the 4 different sensors that are pre-programmed to be connected to a multiplexer and saving the data to the SD card. At the point of writing this, the project is able to save the data in SD card but can't share them unless the SD card is ejected. Furthermore, There are 2 personally designed PCB's for which I will share the design as figures. For further improvements we are looking into adding a timer to control the frequency of the data collected,sending data via LTE-m and using a solar panel for a more consistent and renewable power supply.

To my parents, for their love and support.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CHAPTER 1

PREPARATIONS

1.1 Required Items

Module	Unit Price	Quantity	Total
Stacking Header	1.25	3	3.75
Adafruit Feather M0 Basic Proto - ATSAMD21 Cortex M0	19.95	1	19.95
Adalogger FeatherWing - RTC + SD Add-on For All Feather Boards	8.95	1	8.95
SparkFun Triad Spectroscopy Sensor - AS7265x (Qwiic)	64.95	1	64.95
Melexis Contact-less Infrared Sensor - MLX90614 3V	15.95	1	15.95
RGB COLOR SENSOR WITH IR FILTER	7.95	4	31.8
TCA9548A I2C Multiplexer	6.95	1	6.95
Adafruit BME280 I2C or SPI Temperature Humidity Pressure Sensor	14.95	1	14.95
3.7V 6600mAh Lithium Ion Battery Pack	24.95	1	24.95
CR1220 Coin Battery*	6.95	1	6.95

Adafruit Perma-Proto Breadboard PCB[1]	4.5	1	4.5
22 and 20 AWG stranded copper wires	NA	∞	NA

1.2 Preamble & IDE setup

If you have absolutely no experience with soldering or Arduino, I recommend these guides first.

- Soldering Headers
- Setting up Arduino IDE (Feather M0 proto)
- Tinning wires
- Cutting traces between jumper pads

These are the libraries that we will be using to write our code in:

- Sparkfun Spectral Triad AS7265X
- Adafruit MLX90614
- Adafruit BME280
- Adafruit TCS34725
- RTClib
- SDFat
- SPI[2]
- SD[2]

[1] You can use the quarter or half sized board too.
[2] These libraries may come preinstalled in some arduino setups.

After you've downloaded all the required libraries, try compiling the IDE sketch (Given in the chapter 3). Sometimes you will have some conflicting libraries that you will have to remove through the command line. Your best solution is to google a forum response with either Adafruit/Sparkfun. Stackoverflow is also a good place to find solutions. Once you complete the IDE setup, you can move on to construction.

CHAPTER 2

CONSTRUCTION

2.1 Setting up the Feather M0

- Solder the stacking header onto Feather M0 Proto.(Its preferred that you use a breadboard to solder inorder to align them perfectly.)
- Add the Cr1220 button battery to the battery slot in the Adalogger.
- Cut the trace of the SDCS on the Adalogger. (Please check the guide in 1.2)
- Cut and strip a 22 AWG stranded wire and solder it between SDCS pin and 13 pin on the Feather Adalogger, as shown below. Pin 13 is not printed out on the Adalogger, so always double check the pin number on the Feather M0 Proto.
- Solder the stacking header onto Feather Adalogger now.

2.2 Setting up the Sensors and the Multiplexer

- Solder the pin heads to the RGB sensors and BME280 Environment sensors
- Cut the trace of the Power LED on the Sparkfun Triad sensor.
- Cut and strip 4 pieces of 20 AWG solid wire, roughly 9 ft (3m) each.
- Separate the 2 pieces of wires using either tape or some heat shrink tube.
- Solder the 4 wires to 3V3, GND, SDA and SCL pins on the triad. (Use the 2 Wires with heat shrink tube on SDA and SCL)

- Solder the Multiplexer to the ProtoPermea Breadboard (Its preffered that you solder the pins on C and G) [1]
- Solder 2 very small stripped to Vin and GND and connect them to + and - terminal on the Permea board. Solder 2 more longer wires to the + and - Terminal and connect them to 3V and GND on the Feather M0.
- Solder 3 equally long wired to A0,A1,A2 and connect them to the A0, A1, A2 in the Feather M0.
- Solder 2 stripped wires to SDA and SCL on the Multiplexer and connect them to the SDA and SCL pins on the feather.
- Solder a 12 pin stacking header on the side with 6 SDA and SCL's side. This will be used to connect the sensor pins and multiplexer.
- Solder 2 more stacking headers on the + and - terminals to power the sensors.
- We will now solder the Triad sensors to the PermeaBoard. Solder the SDA and SCL wires to SD0 and SC0/SD1 and SC1. Now Solder the 3V3 and GND pin to + and - respectively.
- Solder some wires to SDA, SCL, 3V/Vin and GND on the sensors. Dont solder the other ends to the PermeaBoard so that we can make it modular. Connect them to the stacking headers.

[1] The steps change if you have printed the personalized PCB. See Appendix A

CHAPTER 3

CODE

3.1 Setup

We are going to setup the code and the libraries in the IDE. Please make sure that you have downloaded the libraries that have been specified in Preamble 1.2. Now we will initialize the libraries. Now each and every sensor has a specified struct and constructor which is expected to be called before using. Since our goal is not just to save the data but to also use it for future we want to make sure that we know which all sensors is it using and at which multiplexer select is it at, we are going to use predefined char array of an optimal length.
Now in the void Setup() loop we will create the testing conditions where, we can test to see if the sensors are responding and what are its I2C values. This part is not really important but can be used for debugging mainly. In the setup we will begin the Wire and Serial library first and foremost, then we see if the SD card is inserted or not and if we can open the filename in the SD card. Now we iterate through the TCAselect from the Multiplexer to see if there is a sensor and if the sensor is found then we look for its I2C to confirm the type of the sensor and save it in an array to be used in the loop().

3.1.1 Psuedocode:

3.2 Triad Sensor

We are going to use the Sparkfun AS7265X library for the Triad sensor. The data is collected in 18 different string values (even if they are actually floating

numbers.). We will have to define the sensor name. Now we for everytime we take a measurement we will have to go like "*sensor*.takeMeasurements()". This takes the measurements from the sensors and stores it in a heap, now to pop the values of the heap we will call the getcalibrated**X**() function where **X** denotes the letters and its wavelength corresponding to it.

Letter	Wavelength	Color (Hex)
A	410nm	#7e00db
B	435nm	#2300ff
C	460nm	#007bff
D	485nm	#00eaff
E	510nm	#00ff00
F	535nm	#70ff00
G	560nm	#c3ff00
H	585nm	#ffef00
R	610nm	#ff9b00
I	645nm	#ff0000
S	680nm	#cc0000
J	705nm	#c90000
T	730nm	#990000
U	760nm	#940000
V	810nm	Infrared
W	860nm	Infrared
K	900nm	Infrared
L	940nm	Infrared

Once we have saved this value in a string we have to go through a few more steps like converting the string into a char and then into an array of char and saving it in the data variables.

3.2.1 Psuedocode:

```
AS7265X sensor;
    sensor.takeMeasurements();
```

```
String a = String(sensor.getCalibratedA(), 2);
String b = String(sensor.getCalibratedB(), 2);
String c = String(sensor.getCalibratedC(), 2);
String d = String(sensor.getCalibratedD(), 2);
String e = String(sensor.getCalibratedE(), 2);
String f = String(sensor.getCalibratedF(), 2);

String g = String(sensor.getCalibratedG(), 2);
String h = String(sensor.getCalibratedH(), 2);
String i = String(sensor.getCalibratedR(), 2);
String j = String(sensor.getCalibratedI(), 2);
String k = String(sensor.getCalibratedS(), 2);
String l = String(sensor.getCalibratedJ(), 2);

String m = String(sensor.getCalibratedT(), 2);
String n = String(sensor.getCalibratedU(), 2);
String o = String(sensor.getCalibratedV(), 2);
String p = String(sensor.getCalibratedW(), 2);
String q = String(sensor.getCalibratedK(), 2);
String r = String(sensor.getCalibratedL(), 2);

char A[9]; char B[9]; char C[9]; char D[9]; char E[9]; char F[9];
char G[9]; char H[9]; char I[9]; char J[9]; char K[9]; char L[9];
char M[9]; char N[9]; char O[9]; char P[9]; char Q[9]; char R[9];

a.toCharArray(A, 9); b.toCharArray(B, 9); c.toCharArray(C, 9);
d.toCharArray(D, 9); e.toCharArray(E, 9); f.toCharArray(F, 9);
g.toCharArray(G, 9); h.toCharArray(H, 9); i.toCharArray(I, 9);
j.toCharArray(J, 9); k.toCharArray(K, 9); l.toCharArray(L, 9);
m.toCharArray(M, 9); n.toCharArray(N, 9); o.toCharArray(O, 9);
p.toCharArray(P, 9); q.toCharArray(Q, 9); r.toCharArray(R, 9);

sprintf(dat0, "Triad, UP, SEN %u,%9s,%9s,%9s,%9s,%9s,%9s ,%9s,%9s,
%9s,%9s,%9s,%9s,%9s,%9s,%9s,%9s,%9s,%9s,%9s",sel, A, B, C, D, E, F,
G, H, I, J, K, L, M, N, O, P, Q, R);
```

3.3 RGB Sensor

For the RGB sensor we are using the TCS34725 Library. When creating the constructor we are going to allot a gain and integration time too. The values from the sensor are saved as a "uint16_t", however the Cpp doesn't support it like C used to do, so we have to convert these into int or float values.

3.3.1 Psuedocode:

3.4 Thermal sensor

The thermal sensor uses the MLX90614 library. The data is saved as a double value. The sensor can take both the object temperature and the ambient temperature.

3.4.1 Psuedocode

3.5 Environment Sensor

The environment sensor is the sensor which is used to measure pressure, altitude, humidty and temperature.

3.5.1 Psuedocode

APPENDIX A

SETTING UP THE FEATHER ON PCB

REFERENCES