Remote I/O Sensor Bus

Akash Sunil Gaikwad (2000116696)
Mohammed Saifullah Sarfraz (2000090709)
Raghavan Naresh Sarangapani (0003440222)

ECE-59500 – Design with Embedded Systems
Electrical and Computer Engineering

Indiana University – Purdue University Indianapolis.
420 University Blvd, Indianapolis, IN 46202
Year: Spring 2017

PROJECT REPORT

Remote I/O Sensor Bus using the FRDM K64f Development Platform

BY
A.S.Gaikwad,
M. S. Sarfraz
R N Sarangapani

Indiana University – Purdue University Indianapolis.
Year: Spring 2017

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Signature:

Name: Akash Sunil Gaikwad (2000116696)

Signature:

Name: Mohammed Saifullah Sarfraz (2000090709)

Signature:

Name: Raghavan Naresh Sarangapani (0003440222)

DATE: 04/26/2017

ACKNOWLEDGEMENT

We would like to express our gratitude and appreciation to all those who gave us the possibility to complete this report. A special thanks to our Project Guide, Professor **Mohamed El-Sharkawy**, whose help, stimulating suggestions and encouragement, helped us to coordinate our project especially in writing this report. given us his full effort in guiding the team in achieving the goal as well as his encouragement to maintain our progress in track.

We would also like to acknowledge with much appreciation of the crucial role of the University library and University Labs, which gave the permission to use all required resources to complete our project.

CONTENTS

- Title Page
- Declaration
- Acknowledgement
- Contents
- Abstract
- Chapter 1: Introduction
- Chapter 2: Requirements for the project
  - 2.1. Software
  - 2.2 Hardware
    - PCB Hardware Design
- Chapter 3: Implementation
- Chapter 4: Result & Conclusion
- Chapter 5: References

ABSTRACT

The goal of this project is to setup a remote monitoring system that can take inputs from various sensors through a common bus design and send the data to cloud for monitoring. The project uses the Hexiwear IoT platform which is based on the FRDM K64F controller. In the scope of this project the values from the voltmeter, ammeter and ambient light sensors are sampled regularly and the same is updated to the cloud using the ESP8266 Wi-Fi module. The Hexiwear docking station is used to connect the sensors, in the form of click boards. The SPI protocol is used to sample values from the Voltmeter and Ammeter sensors, the I2C protocol for ambient light sensor and the UART protocol to transfer data to the ESP8266 Wi-Fi module. The data from the device is transferred using TCP/IP protocol to the Math works Thing speak platform, from where it can be analyzed using visualizations. The future work on this project involves integration of more sensors like pressure and fingerprint sensors, managing the power usage from solar panel and integrating cloud data to a personalized website.

CHAPTER 1

INTRODUCTION

The Hexiwear main board is docked to the Hexiwear docking station, from where the various MikroeElectronica Sensor clicks are interfaced. The click modules used were MIKROE-2436 Voltmeter click, MIKROE-2377 Ammeter click and MIKROE-1903 Ambient light sensors. The Wi-Fi module ESP 8266 board was used to transfer the data from the board to Mathworks Thingspeak cloud. The MIKROE-2436 Voltmeter click uses the OpAmp as a differential amplifier to measure the voltage between the + and – points. This voltage is passed through a 12 bit ADC, which is nearly 33 times scaled down as the input, and is finally read through SPI interface by the MCU. The MIKROE-2377 ammeter click module makes use of a shunt resistor to measure the current being passed at the + and – ends of the probe, in which a proportional voltage is generated across the shunt resistor. This is then passed to the on-chip ADC, and values in the range 1mA to 1A can be read using the SPI interface. The MIKROE-1903 Ambient 2 light click uses TI's OPT3001 sensor, to measure only the visible light spectrum. The intensity of the source light is measured using the transitional impedance as the light incident on the photodiode changes its characteristics of the voltage drop with for a given load current. Finally, this value is passed through an ADC and is read using I2C interface. The ESP8266 Wi-Fi module is an SoC with TCP/IP data transfer capabilities. It can be used both as an access point and as a Wi-Fi client, in this project it is used as a Wi-Fi client and connected to a Wi-Fi network using TCP protocol to transfer the data. The click modules were selected keeping in the power requirements, resolution input/output specs into consideration. The ADC output we measured from the voltmeter and ammeter click modules has a good precision and just 1 % error to true value in some readings as verified with a DMM (Digital Multi-meter).

CHAPTER 2

REQUIREMENTS FOR THE PROJECT

2.1. Hardware

1. FRDM K64F: The Freedom-K64F is an ultra-low-cost development platform for Kinetis K64, K63, and K24 MCUs. It has Form-factor compatible with the Arduino R3 pin layout Peripherals enable rapid prototyping, FRDM K64F board has 6-axis digital accelerometer and magnetometer to create full eCompass capabilities, a tri-colored LED and 2 user push-buttons for direct interaction, a microSD card slot, and connectivity using onboard Ethernet port and headers for use with Bluetooth® and 2.4 GHz radio add-on modules

It offers OpenSDAv2, the NXP open source hardware embedded serial and debug adapter running an open source bootloader, offers options for serial communication, flash programming, and run-control debugging.

FRDM K64F Main board

FRDM K64F hardware description:

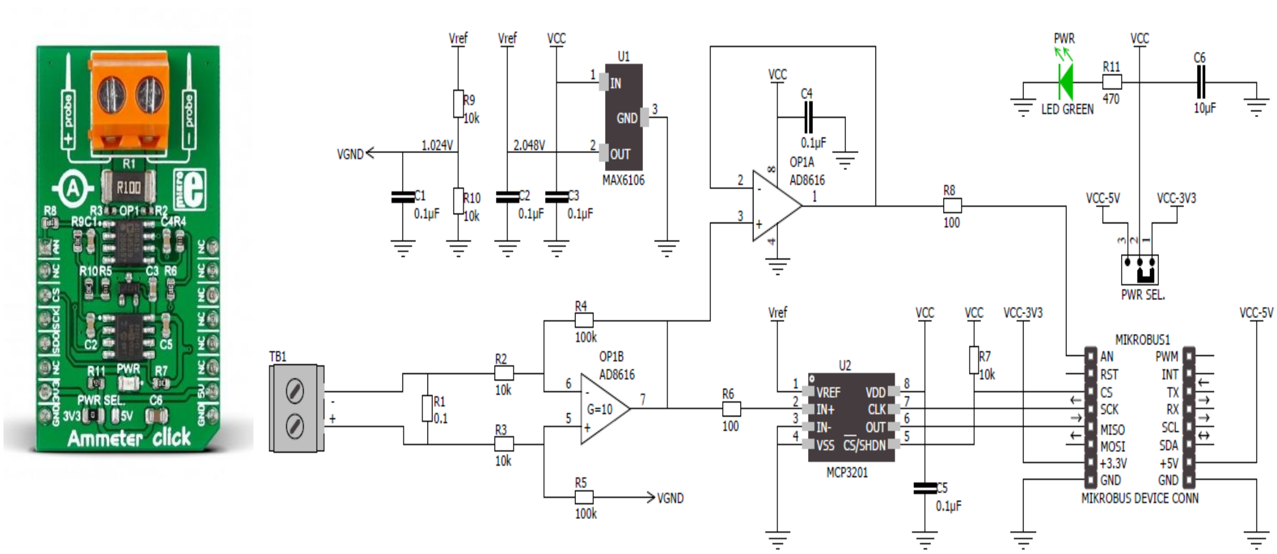
- MK64FN1M0VLL12 MCU (120 MHz, 1 MB flash memory, 256 KB RAM, low-power, crystal-less USB, and 100 Low profile Quad Flat Package (LQFP))
- Dual role USB interface with micro-B USB connector
- RGB LED
- FXOS8700CQ accelerometer and magnetometer
- Two user push buttons
- Flexible power supply option – OpenSDAv2 USB, Kinetis K64 USB, and external source
- Easy access to MCU input/output through Arduino™ R3 compatible I/O connectors
- Programmable OpenSDAv2 debug circuit supporting the CMSIS-DAP Interface software that provides:
    - Mass storage device (MSD) flash programming interface
    - CMSIS-DAP debug interface over a driver-less USB HID connection providing run-control debugging and compatibility with IDE tools
    - Virtual serial port interface
    - Open source CMSIS-DAP software project
- Ethernet
- SDHC
- Add-on Bluetooth module: JY-MCU BT board V1.05 BT

2. Ammeter Click Module:

The Ammeter click module uses MCP3201 ADC, it is 12-bit Digital Analog to Digital Convertor (ADC) The MCP3201 features a successive approximation register (SAR) architecture and an industry-standard SPI serial interface, allowing 12-bit ADC capability to be added to any microcontroller. Ammeter click module consist of MCP3201 ADC, MAX6106, AD8616 OpAmp On board module. MAX6106 is low-cost, low-dropout (LDO), micro power voltage references module, AD8616 is amplifier featuring very low offset voltage, wide signal bandwidth.

This Click module can measure maximum voltage up to 48 V and Measures both DC and AC, 0.1A - 1A measurement range. It requires 3.3V or 5V power supply for operation. It has Two onboard screw terminals (probe+ and probe-) are bringing in the current, which then passes through a shunt resistor. A voltage proportional to the strength of the current is generated across the resistor. This voltage passes through an amplifier before it's fed into a 12-bit ADC which then outputs a digital value through the SPI interface.

The code processes the digital value to determine the exact current between 1mA and 1A for DC current. It's also possible to measure AC current by deriving the value from peak to peak measurements.



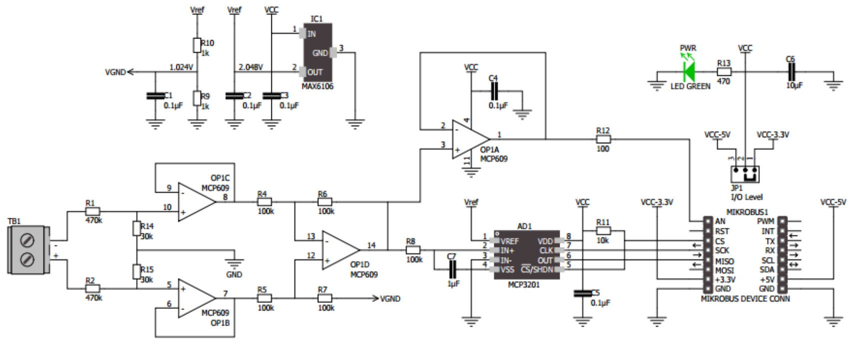
Schematic of the ammeter click module

3. Voltmeter Click Module:

The design of the board is based on an OpAmp set up as a differential amplifier with a buffered input. Two onboard screw terminals (probe+ and probe-) are bringing in the current, which then flows through a row of four resistors. A voltage – proportional to the input charge – is generated across the last two resistors. From there it is sent to the differential amplifier that further intensifies the difference between the two inputs (+/-).

The resulting charge is exactly 33 times lower than the actual measured voltage. It is converted by an onboard 12-bit ADC before being outputted through the mikroBUS™ SPI interface (the firmware in the target board MCU should be set up to multiply the ADC value to get the actual voltage). It is also possible to output the analog value directly through the mikroBUS™ AN pin (useful if the target MCU has a higher resolution ADC).

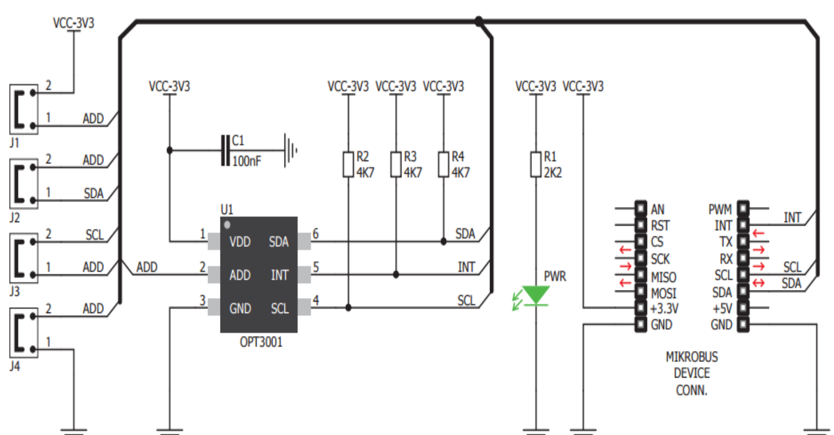
The board can work with either a 3.3V or a 5V power supply.



Schematic of the Voltmeter click module

4. Ambient light Click Module:

Ambient 2 click carries TI's OPT3001 Ambient Light sensor. It's a small (2mm x 2mm) single-chip lux meter that measures only the visible part of the light spectrum from any kind of source

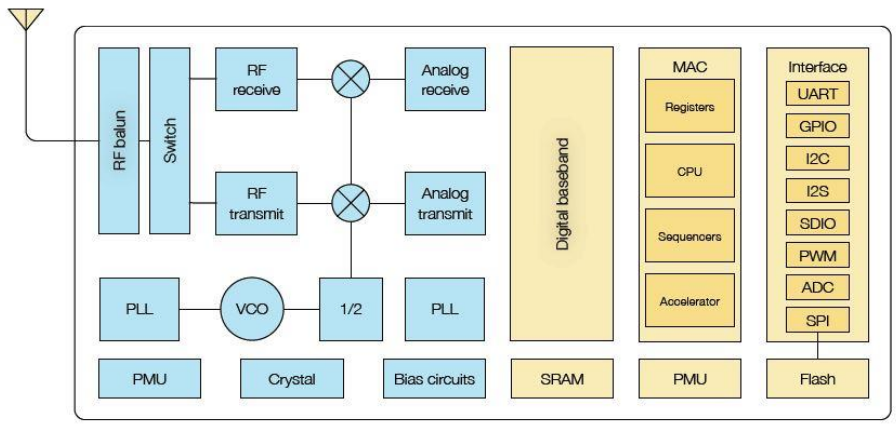


(mimicking the way humans see light). It does so by filtering out 99% of infrared light. The measurement range of Ambient 2 click is from 0.01 Lux to 83k lux, and a 23-bit resolution. Communication with the target MCU is done through mikroBUS I2C pins, with an additional INT pin which can be used for triggering wake-up events (offloading the MCU).

Schematic of the Ambient light Click Module

5. Wi-Fi Module ESP8266:

The ESP8266 Wi-Fi Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your Wi-Fi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor. Each ESP8266 module comes pre-programmed with an AT command set firmware.



Functional Block Diagram

Features:

- 802.11 b/g/n
- Wi-Fi Direct (P2P), soft-AP
- Integrated TCP/IP protocol stack
- Integrated TR switch, balun, LNA, power amplifier and matching network
- Integrated PLLs, regulators, DCXO and power management units
- +19.5dBm output power in 802.11b mode
- Power down leakage current of <10uA
- 1MB Flash Memory
- Integrated low power 32-bit CPU could be used as application processor
- SDIO 1.1 / 2.0, SPI, UART
- STBC, 1×1 MIMO, 2×1 MIMO
- A-MPDU & A-MSDU aggregation & 0.4ms guard interval
- Wake up and transmit packets in < 2ms
- Standby power consumption of < 1.0mW (DTIM3)

6. Digital Multi Meter (DMM):

A digital multimeter (DMM) is a test tool used to measure two or more electrical values—principally voltage (volts), current (amps) and resistance (ohms). It is a standard diagnostic tool for technicians in the electrical/electronic industries. We use the DMM to measure the voltage and current and verify it with the readings from our sensors.

Digital Multi Meter

7. DHT 22 Temperature and Humidity Sensor

DHT22 Click is a temperature and humidity measurement board carrying the AM2302 sensor which communicates with the desired microcontroller through a single serial data line.

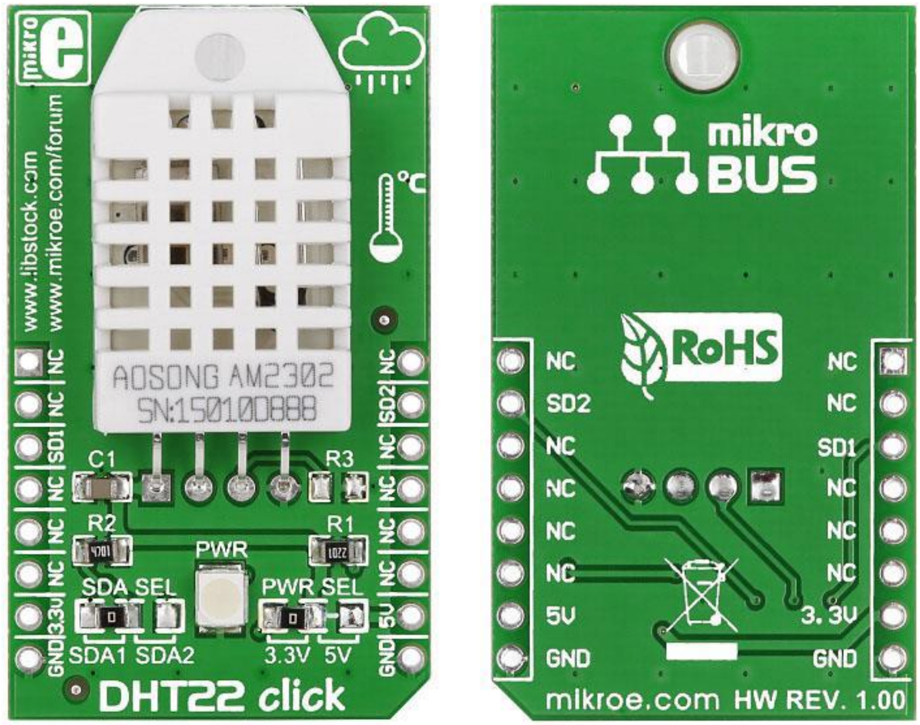
Interfacing Steps:

• The DHT22 has a single serial data line.

• There is an option to choose between the CS (Chip Select) or the INT (Interrupt) pins for communicating.

• The SDA SEL pins allows the selection of either CS (SDA1) or INT (SDA2) for the preferred wire signal.

• Connect the CS(SDA1) Pin on the module to the CS(PTC2) Pin on the FRDM K64F.



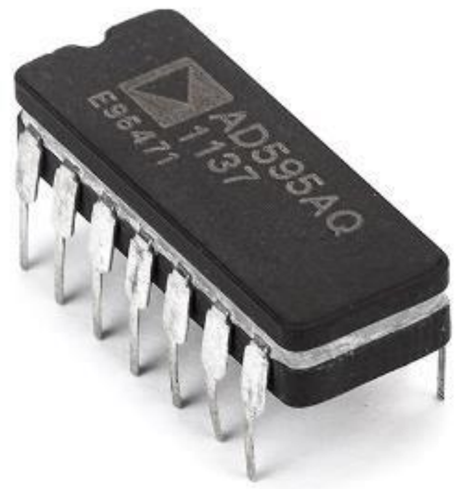
DHT Sensor

8. AD595 Analog Thermocouple Sensor IC:

AD595 is an instrumentation amplifier and thermocouple cold junction compensator which produces a high level of output which is 10 mV/°C from a thermocouple signal. The AD595 is powered by a single ended supply of +5 V with a low supply current of 160 µA. AD595 is a 14-pin DIP packaged IC with a calibration accuracy of ±1°C. The AD595 IC is a type K thermocouple which gives a differential voltage (mV) corresponding to the Type K voltage depending on the thermocouple temperature at that instant. It is an analog IC and it is connected to the Hexiwear Brain via an analog pin PTB3.

Interfacing Steps:

- Analog IC595 has the input of 5V at the 11th pin.
- Connect ground at pins 1,4 or 7.
- The thermocouple should be connected between pins 1 and 14.
- The pins 8 and 9 are shorted and connected to the FRDM – K64F at PTB3.



AD595 Thermocouple

7. DC motor As Load:

DC motors are so called because they are powered by a current placed across the electrodes of the motor. The speed rating of a DC motor is the top speed it can run at. The actual speed the motor runs at is a function of how strong the current is that is applied to the motor. We use this motor as a load and then measure the current in series with the load.

DC Motor

8. DC Power Supply:

A DC power supply is one that supplies a constant DC voltage to its load. Depending on its design, a DC power supply may be powered from a DC source or from an AC source such as the power mains. We use the power supply to give the voltage input for the voltmeter and ammeter measurements.

DC Power Supply

8. Finger print Sensor

- On-board optical sensor and 32-bit CPU.
- After the corresponding command, finger print is scanned 3 times to form a Template.
- The Fingerprint Scanner compares the 32 Bit User input with Template.
- The on-board JST-SH connector has four signals: VCC, GND, TX, RX.
- The Memory can hold 20 different fingerprint templates.
- The Templates can be downloaded/uploaded using the serial interface.

Finger Print Sensor

Hardware Design

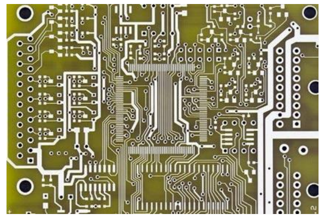
As in previous part of project, we used only three sensors. But as per company requirement, they want multiple sensors to be interfaced with the FRDM K64F, We have around 8-9 sensors. The Hexiwear Docking station has only 3 sensor mounting port. That's why we decided to design our own PCB,

We used Eagle PCB software to design the PCB.

- **EAGLE** is a scriptable electronic design automation application with schematic capture, printed circuit board layout, auto-router and computer-aided manufacturing features.
- We used EAGLE 8.1.1 Commercial License version.
- In Free Eagle version, We can use up to only 80 cm*cm Area for Manufacturing purpose.
- We have used 160 cm*cm area to design our PCB.

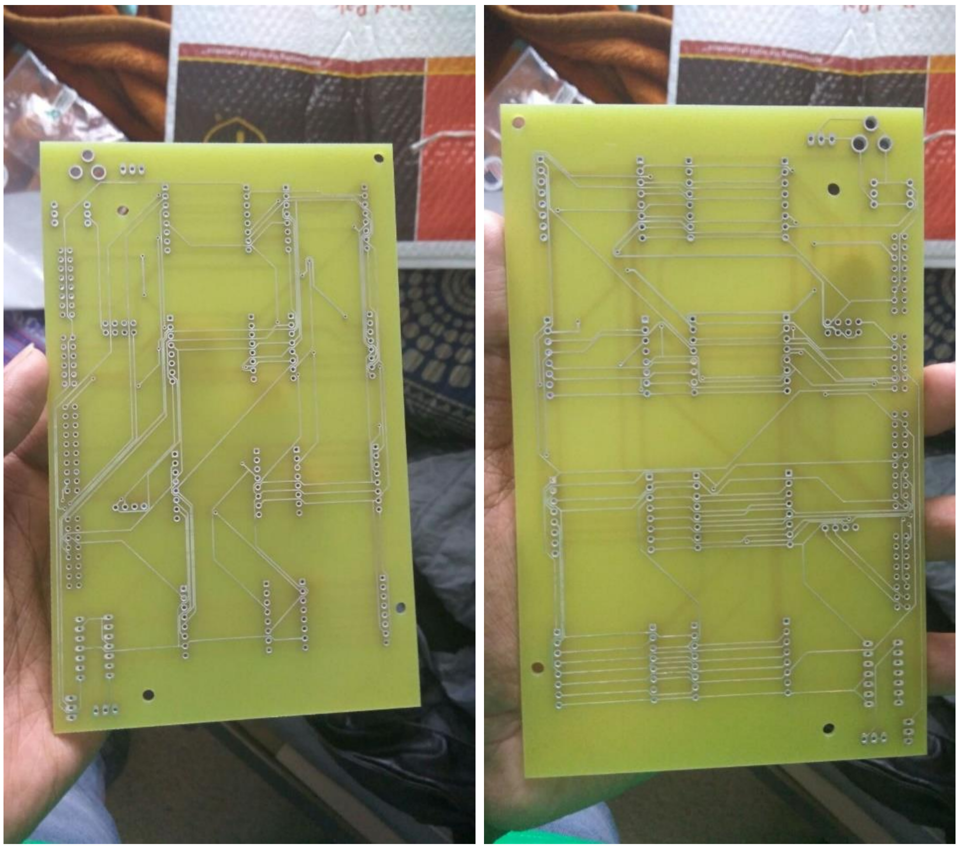
What is Barebones PCB?

- "Bare Bones PCBs" is simple 2 layer PCBs that do not have Solder Mask and Top silk Screen.
- Such PCBs is useful when you have simple applications for your PCB,
- We can hand-solder simple devices on board.
- Its like Zero PCB with Connection.
- It is mainly used to Test the PCB Design
- We have 2 layer PCB Design.

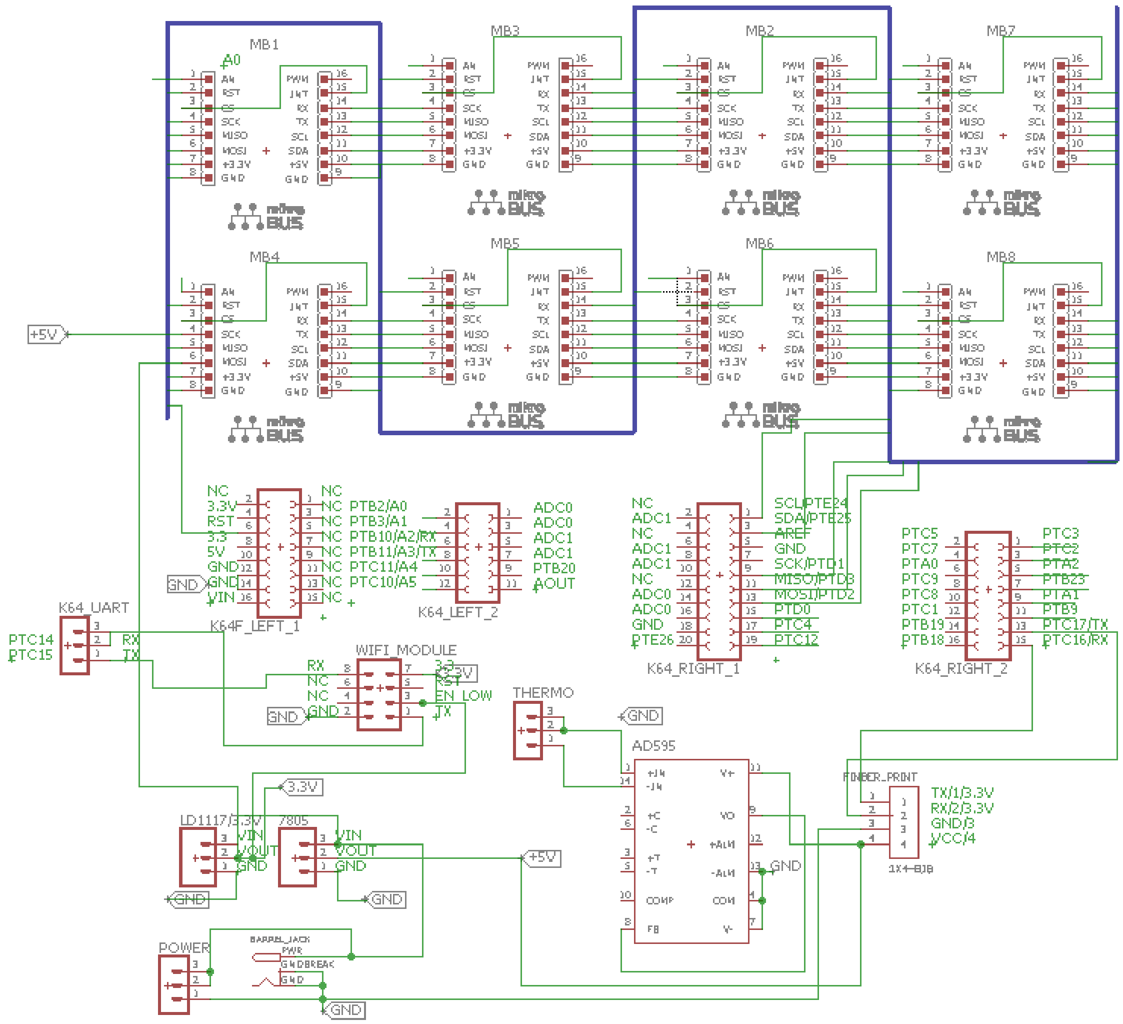


Bare Bone PCB

PCB after Manufacturing

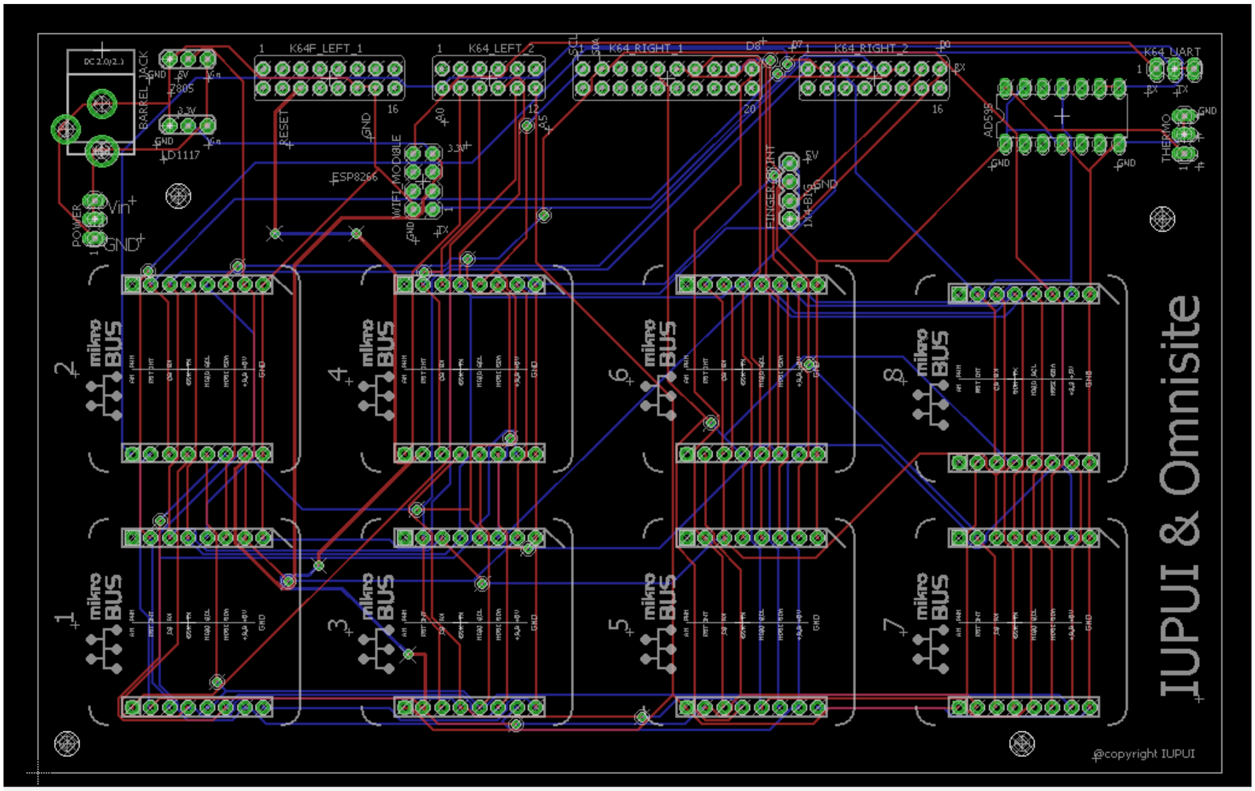


Schematic Diagram of PCB (.sh File)



(.sh File)

Actual PCB layout (.brd File)



(.brd File)

- Blue lines indicate 1st layer of Connections
- Red Lines Indicates 2nd layer of Connection
- Small Circle indicates the Via (Inter connection between 1st and 2nd layer)

Manufacturing of PCB

- To manufacture any PCB, we have to provide Gerber Files to the manufacture.
- The **Gerber** format is an open ASCII vector format for 2D binary images.
- We used following Gerber files to manufacture the PCB.

Top Silkscreen: Akash.GTO

Top Soldermask: Akash.GTS

Top Copper: Akash.GTL

Bottom Copper: Akash.GBL

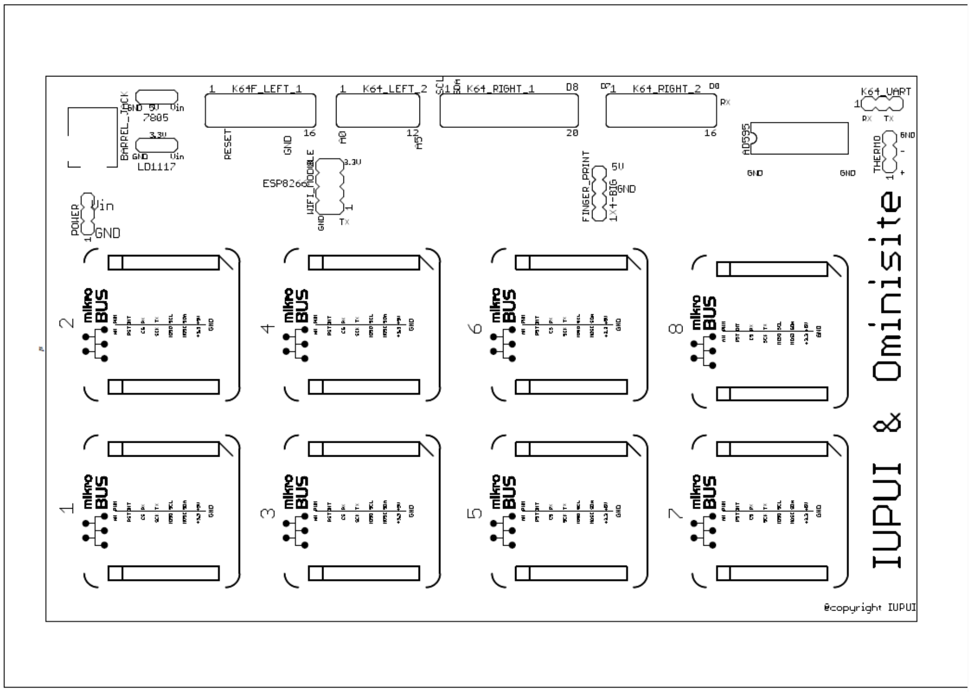
Bottom Soldermask: Akash.GBS

Drill: Akash.TXT

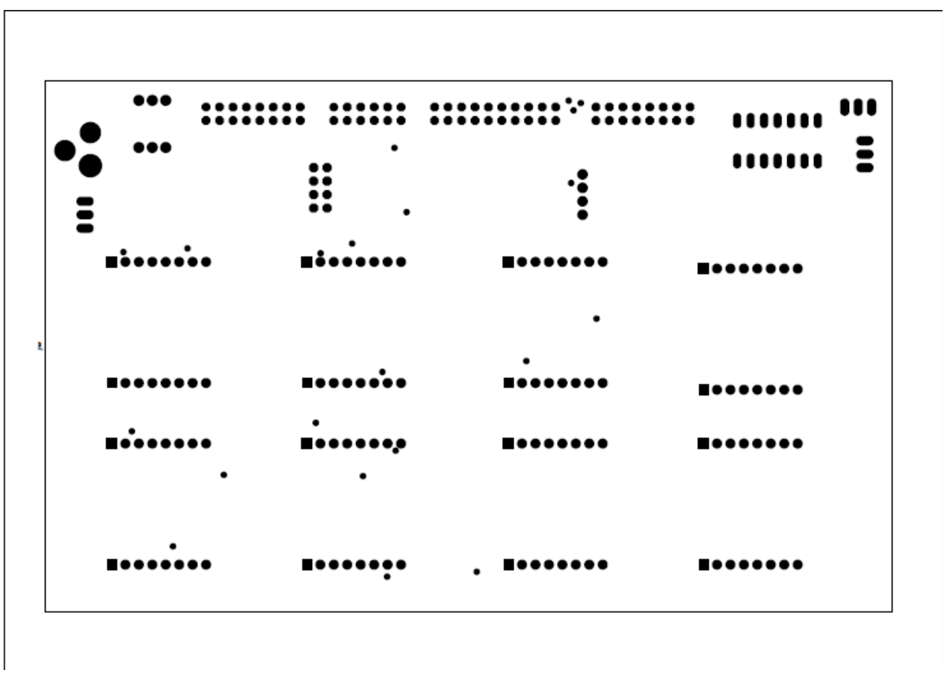
Content of Gerber Files

Different layers of PCB

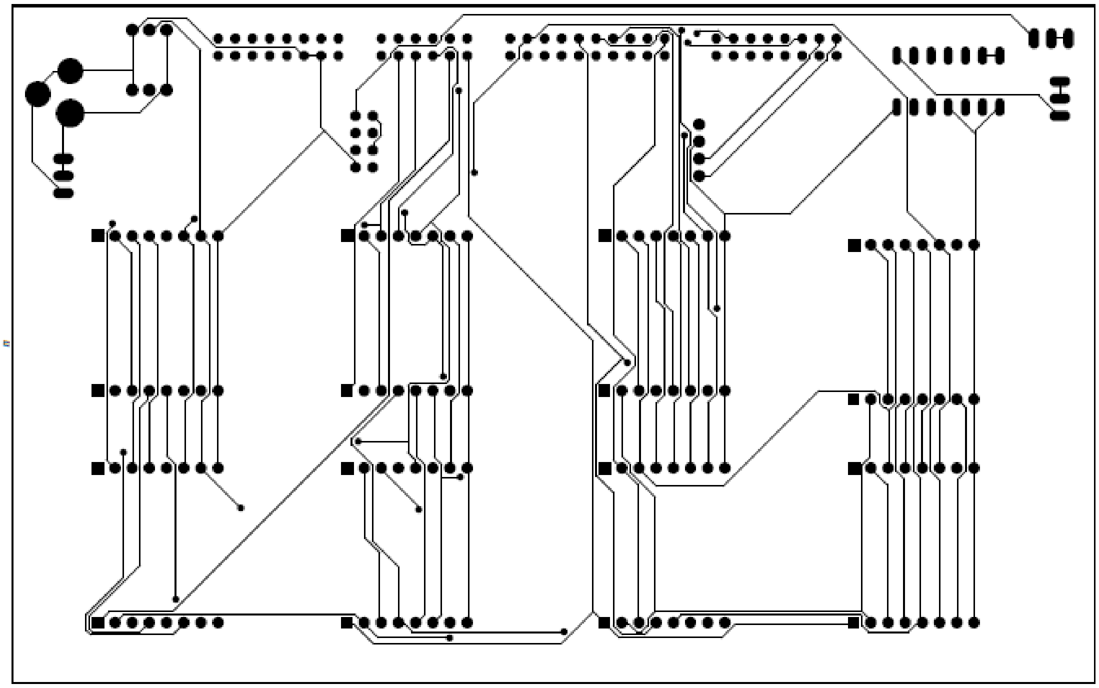
1. **Top Silk Screen:**



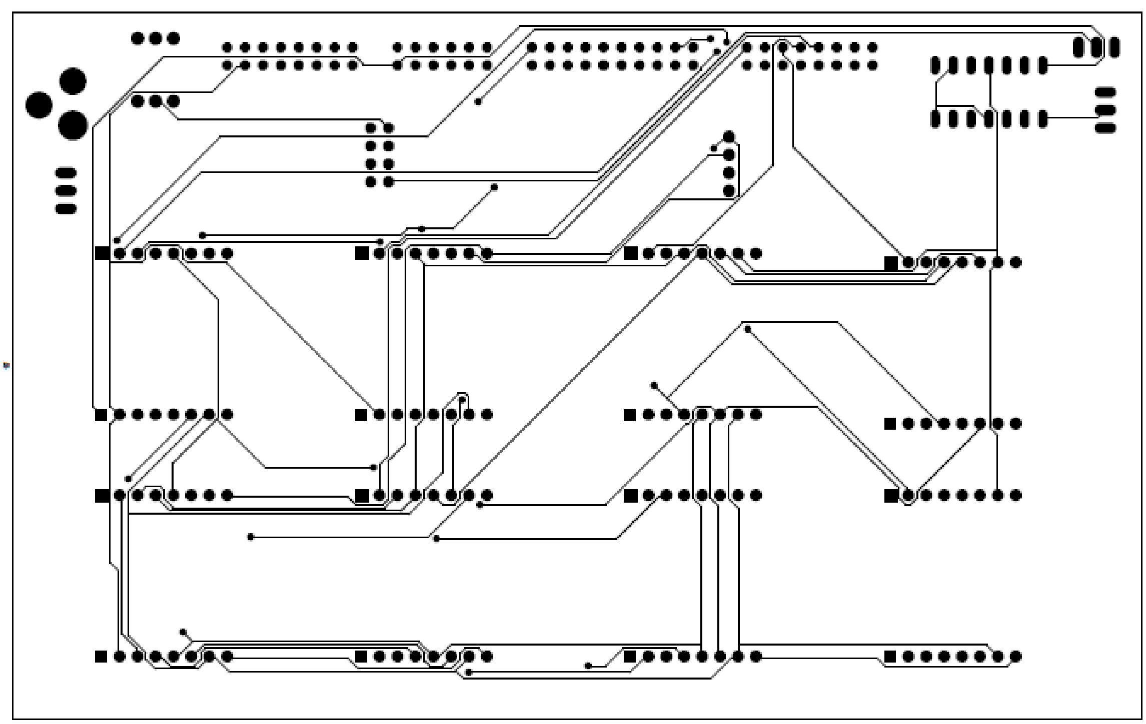
2. **Top Solder Mask:**



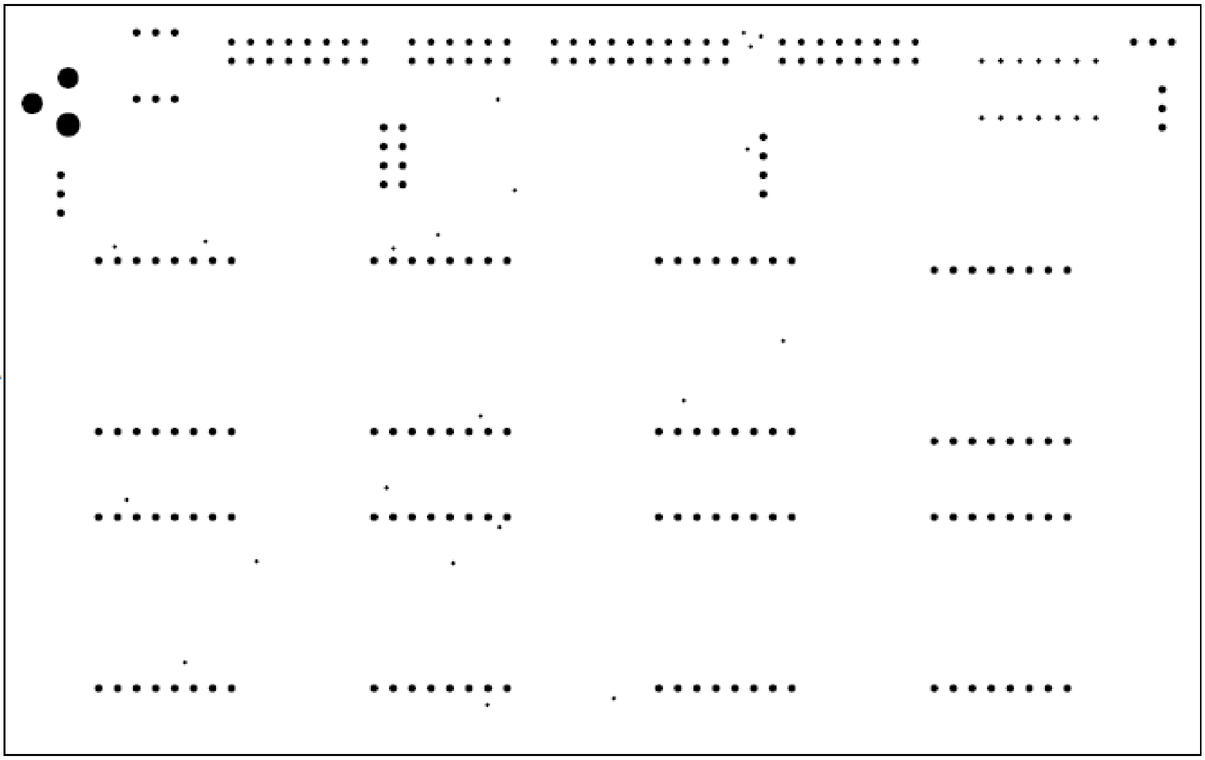
3. **Top Copper Layer:**



4. **Bottom Copper Mask:**



5. Drill File:

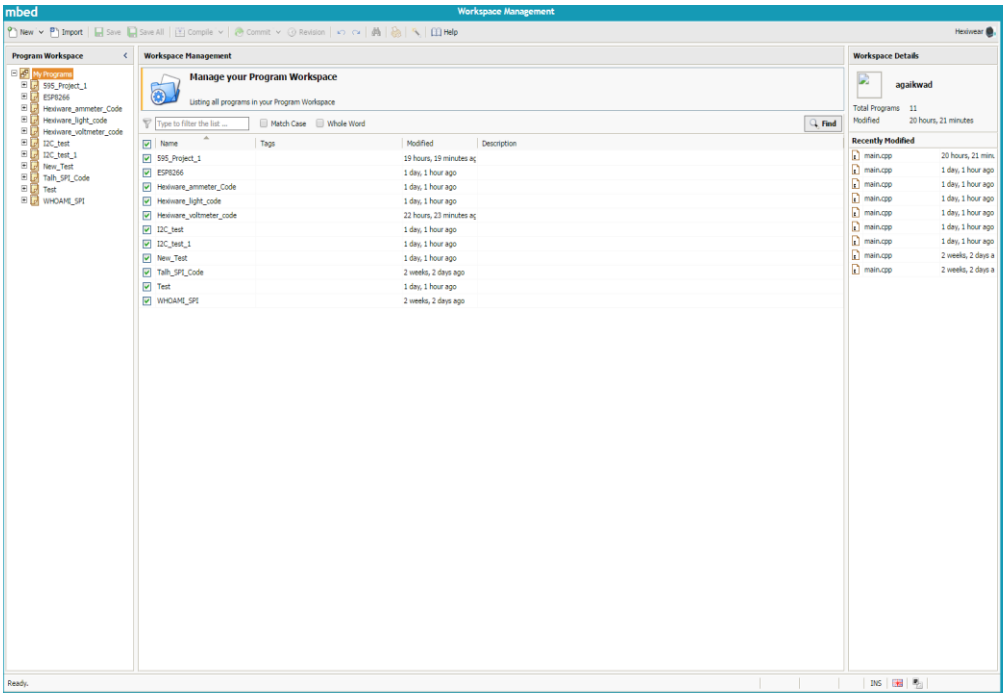


PCB Manufacturing Solution:

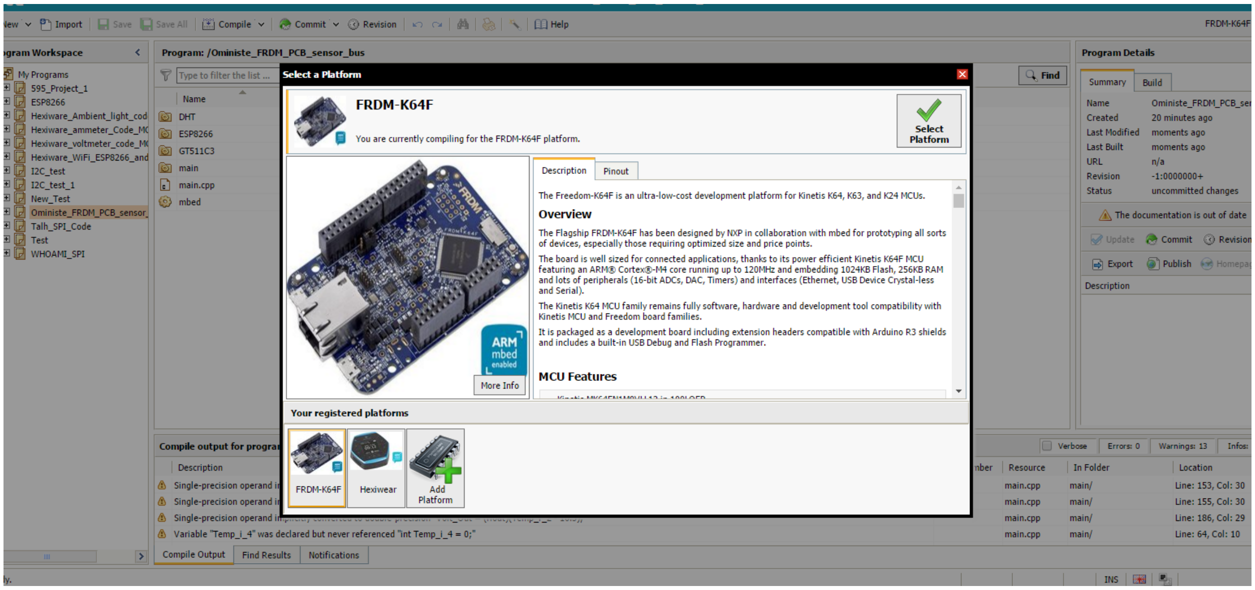
- Advance Circuit
- Basic PCB
- Element 14
- San Francisco Circuits
- Royal Circuits

2.2 Software

The mbed OS is used to write applications that run on embedded devices, by providing the layer that interprets the application's code in a way the hardware can understand. Application code is written in C++. It uses the *application programming interfaces* (APIs) that mbed OS provides. These APIs allow your code to work on different microcontrollers in a uniform way. This reduces a lot of the challenges in getting started with microcontrollers and integrating large amounts of software.



Mbed coding screen



Mbed FRDM K64F platform selection

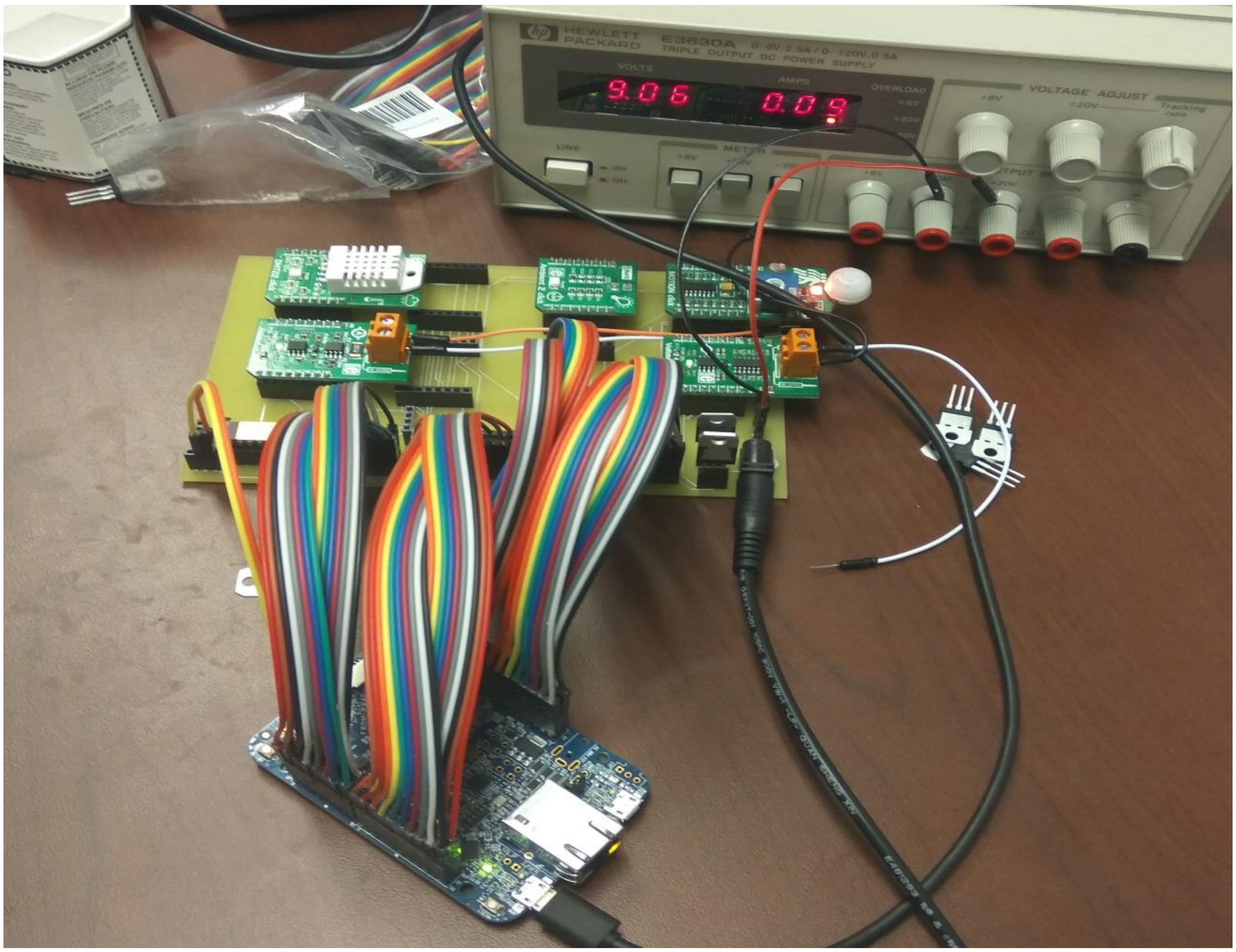
CHAPTER 3

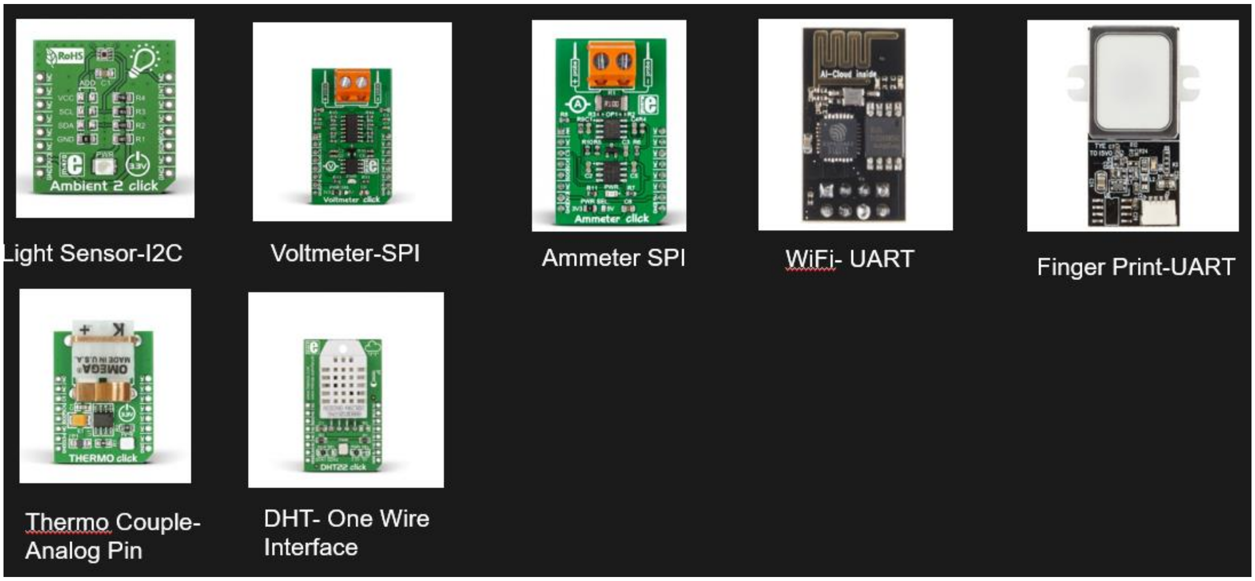
IMPLEMENTATION

The Implementation of the project is done with the following steps.

1. Board connection

Plug the all the sensor on PCB and connect the FRDM board PCB, Plug USB cable to your computer and the other end to the micro-USB port of the Docking station. Make sure that the PWR switch of the Docking station is set ON and the green PWR status light will come on, indicating it has power. After a few seconds of activity, the PC will recognize the mbed Microcontroller as a standard USB drive called "Mbed".





Sensors Used

1. The DC motor is connected in series with supply and ammeter to measure the current.
2. The Ammeter click is connected to Port 1. SPI interface via PTC6- MOSI, PTC7-MISO and PTC5-CLK and Chip Select via PTC4 pin.
3. The FRDM K64f brain is connected to the respective port. From here all the interface lines are taken in the docking board.
4. The Voltmeter click is connected to Port 1. SPI interface via PTC6- MOSI, PTC7-MISO and PTC5-CLK and Chip Select via PTC5 pin.
5. The Power supply is divided between the Voltmeter, Ammeter and the Wi-Fi chip.
6. The Ambient light sensor is connected through the breadboard because it uses only the I2C lines from the docking board port 3.
7. The ESP8266 Wi-Fi module is connected by using the UART lines from the same port 3 of docking station.

2. Source code:

The Source code uses the following procedures:

- The SPI interface is initialized for Voltmeter and Ammeter clicks, the I2C protocol for Ambient light and the UART protocol for the Wi-Fi chip.
- The ESP 8266 chip is initialized using the commands RESET and AT. Once initialized the device is connected to Access point using AT+CWJAP using SSID and Password. Finally the MUX is selected for a single channel transfer.

- Next the Ammeter ADC data is read from the SPI. This data is shifted such that the lower 7 bytes are taken and 5 bytes are taken from higher byte. The final value is calculated using the formula.

  Step_output = ((ADC_Value * 1) / 4096)

  Ammeter_output = ((Step_ouput - 0.5) * 1000)

- After this the Voltmeter ADC data is read from the SPI. This data is shifted such that the lower 7 bytes are taken and 5 bytes are taken from higher byte. The final value is calculated using the formula.

  Step_output = ((ADC_Value * 33) / 4096)

  Voltmeter_output = (Step_ouput - 16.5)

- The Ambient light works with I2C protocol, first the configuration register 0x01 is written with 0xCC01 value. Next the values are read from the register 0x00.

  Step_output_1 = Read_Low_Byte >> 4

  Step_output_2 = (Read_Low_Byte - (Step_output_1 << 4)) *256 + Read_High_Byte

  Light_output = (Step_output_2*l)/100

- The Thermocouple gives the output through the Analog pin, the 16 bit values is calculated according the following formula to get the voltage difference.

  Temp_f_1 = AN_Thermo.read_u16()

  Temp_f_1 = ((Temp_f_1 / 65536) * 330)

- The Motion module works with the Analog pin. If the pin is high it indicates motion and if it is low it indicates no motion. The range of the motion detection can be adjusted using the potentiometer in the sensor click.

- The DHT sensor gives the accurate measurement for the temperature and humidity. It can be obtained using the library function as shown below. The temperature is measured in Fahrenheit and the humidity in dew point units.

  Temp_f_1 = DHT_Temp_Hum.ReadTemperature(FARENHEIT);

  Temp_f_2 = DHT_Temp_Hum.ReadHumidity();

- Next this value is sent by forming a TCP packet with the Write Key and individual field values for each field. After sending this in TCP we check the Acknowledgement and confirm it in the Thingspeak portal.

"GET
https://api.thingspeak.com/update?key=45IE5JPN8ZIK2W2M&field1=%f&field2=%f&field3=%f\r\n",ammeter_out,voltmeter_out,light_out

- The security is implemented using the fingerprint sensor. When a motion is detected in the motion module or when a finger is pressed in the sensor, the options are displayed on the screen. The following options were implemented.
  - **Verify**:
    Used to verify the fingerprint. If authenticated shows the ID in which it was stored and displays the following sensor values. If not authenticated, hides the sensor values on the serial port.
  - **Enroll**:
    Used to enroll the fingerprint. When selected will request a password to be entered in the terminal, only on authentication with password the enrollment begins. Next a free ID is identified from the sensor. Enrollment needs finger to be placed three times. The sensor is programmed to turn On and Off three times to get the fingerprint from the user. Once the enrollment is successful the verification can be done. If ID exceeds 20 then user needs to delete an ID to enroll.
  - **Delete:**
    Has options implemented to delete specific ID or all ID's completely. Implemented a password authentication for added security. Has measures to check if all ID's are deleted.

3. Loading the program:

The source code for this program is written and saved to the DAP-LINK in the file explorer done in a similar way as that of an USB drive. This is called program flash. When the Reset_K64F Button located on the Docking station is pressed, the newest program on the DAP-LINK drive will be loaded in to the K64F microcontroller FLASH memory.

When the program has been loaded onto the K64F microcontroller, it will then start running.

4. Download the program:

Download a (.bin) to FRDM K64f

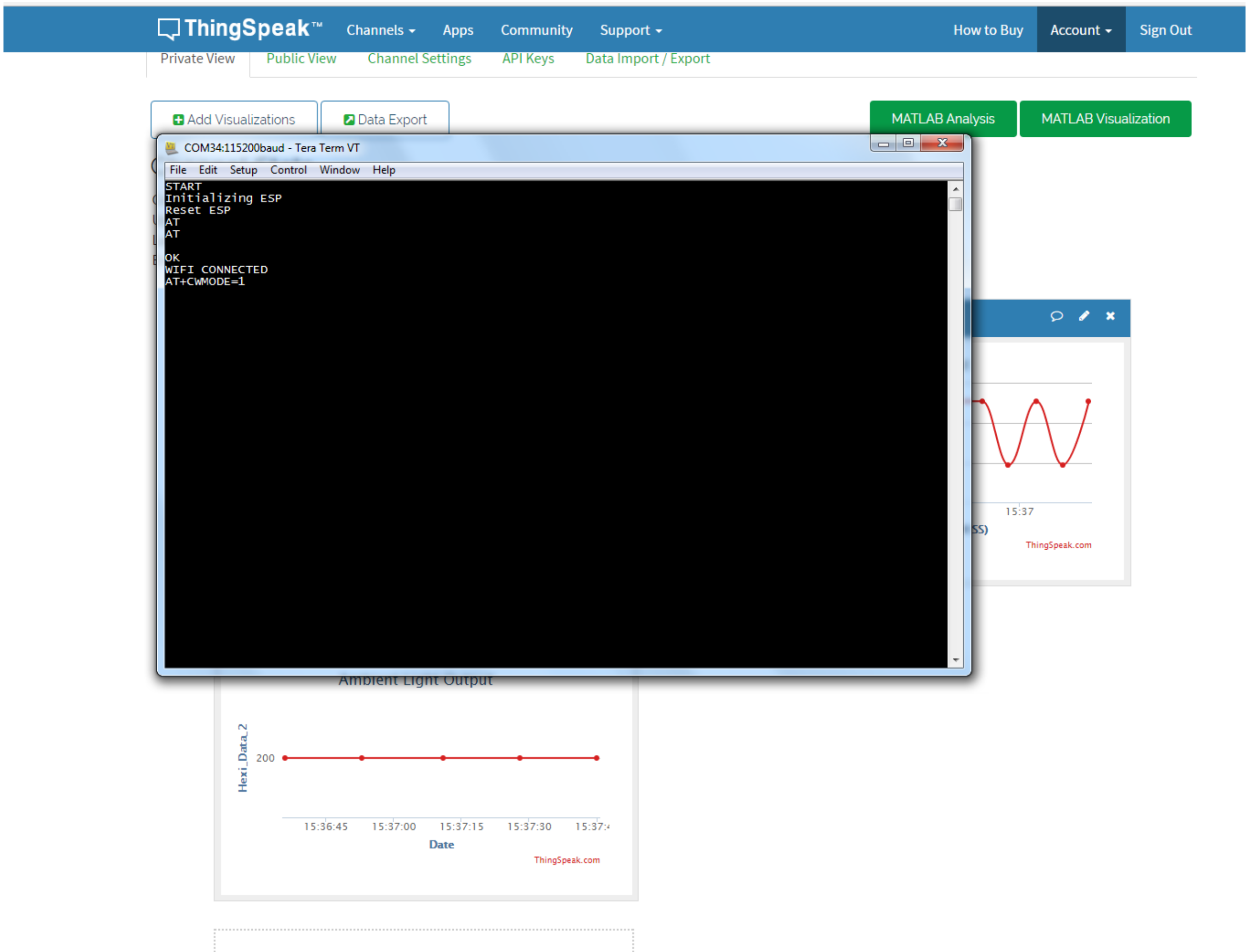
Make sure the power is given to the bus PCB.

Press the Reset Button

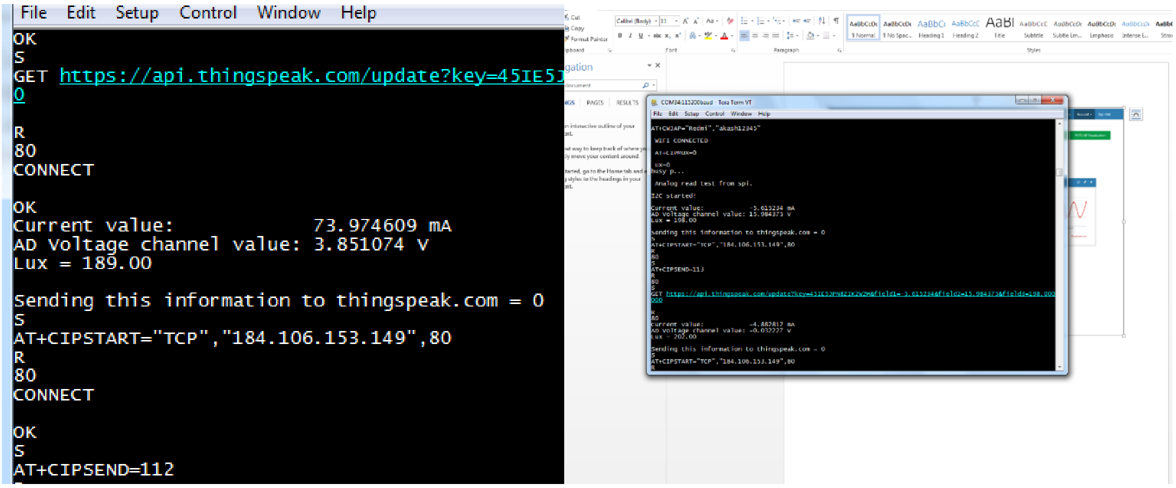
Run the Program.

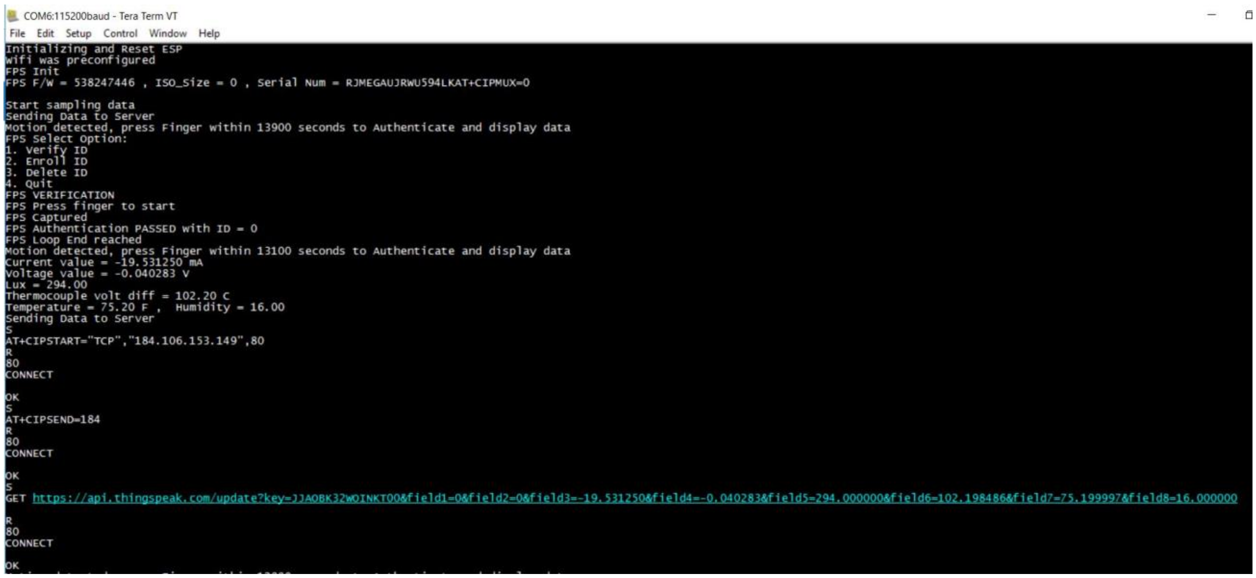
CHAPTER 4

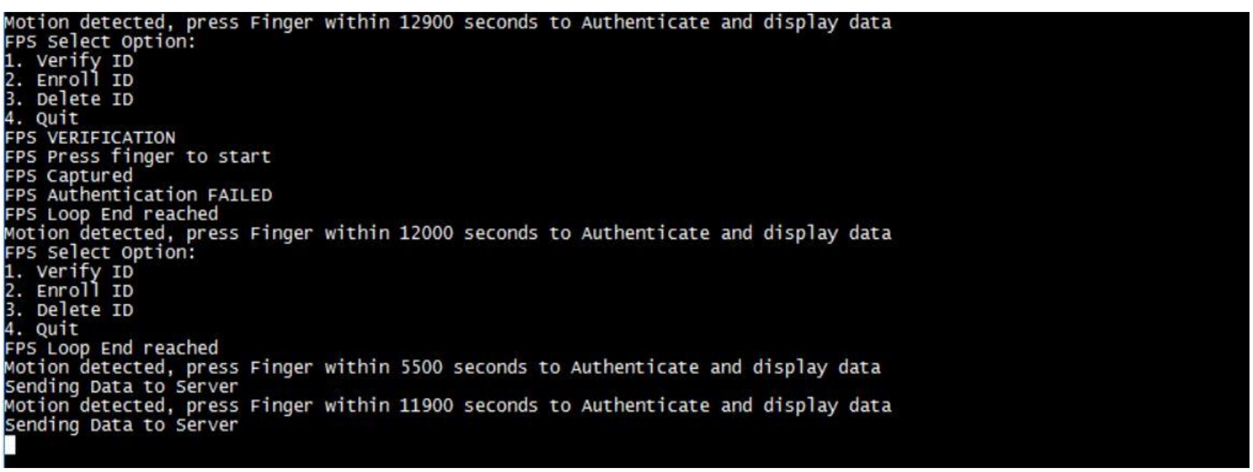
RESULT & CONCLUSION

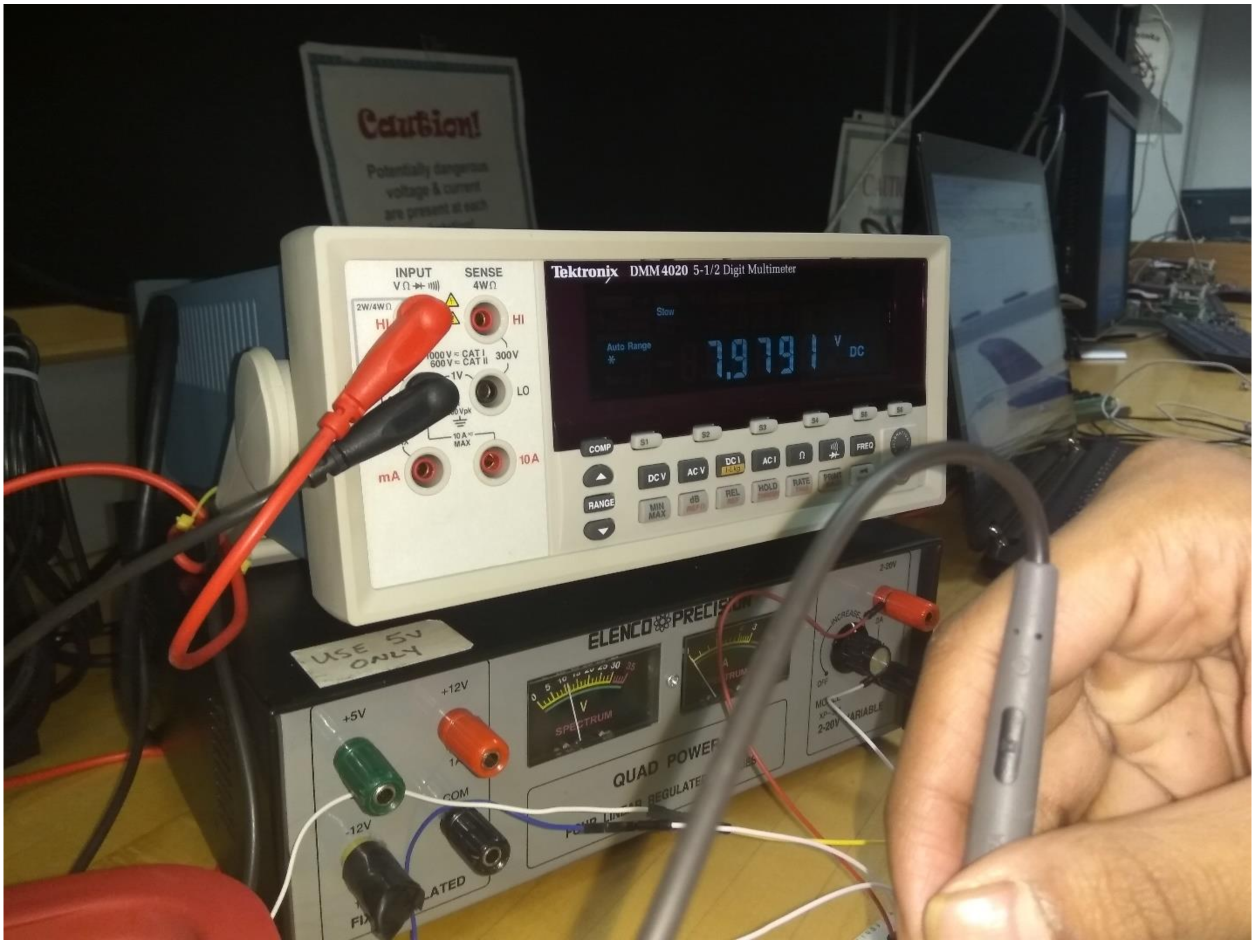


WiFi Connection Established

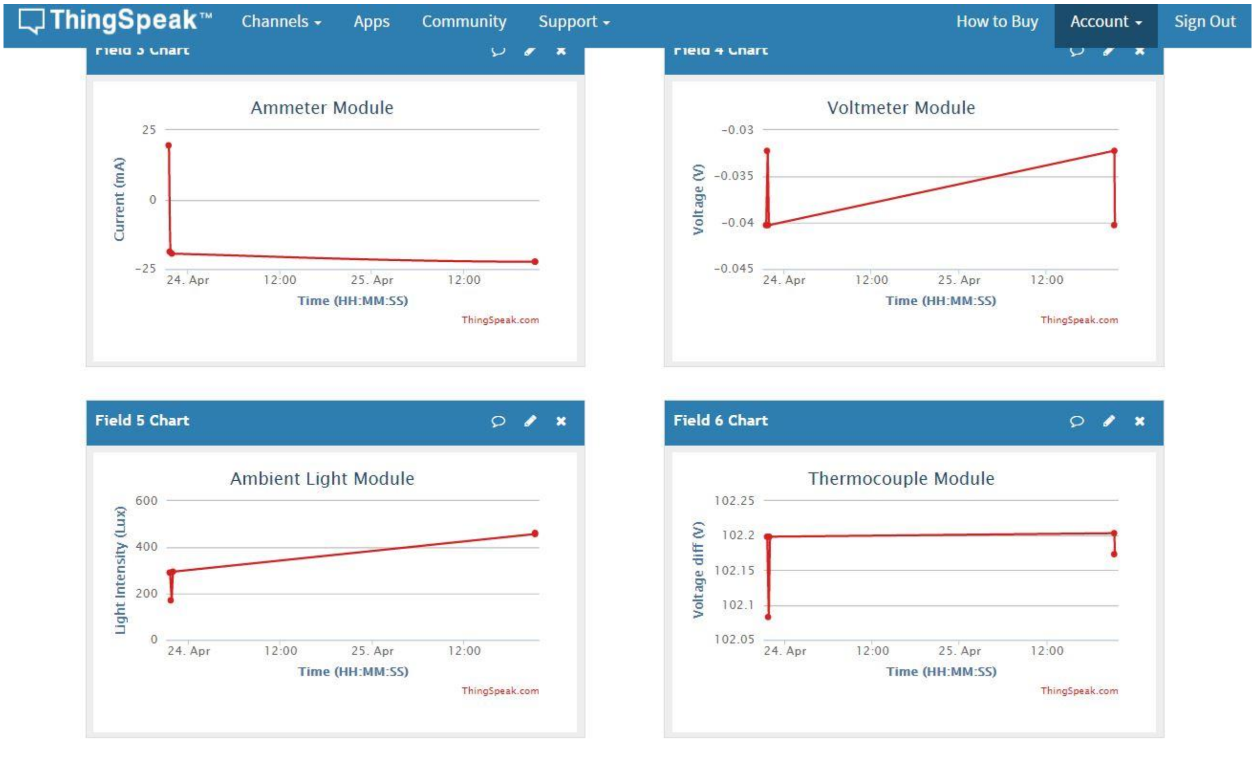

File Edit Setup Control Window Help
OK
S
GET https://api.thingspeak.com/update?key=45IE5J
0
R
80
CONNECT
OK
Current value: 73.974609 mA
AD Voltage channel value: 3.851074 V
Lux = 189.00
Sending this information to thingspeak.com = 0
S
AT+CIPSTART="TCP","184.106.153.149",80
R
80
CONNECT
OK
S
AT+CIPSEND=112


COM6:115200baud - Tera Term VT
File Edit Setup Control Window Help
Initializing and Reset ESP
wifi was preconfigured
FPS Init
FPS F/W = 538247446 , ISO_Size = 0 , Serial Num = RJMEGAUJRWU594LKAT+CIPMUX=0
Start sampling data
Sending Data to Server
Motion detected, press Finger within 13900 seconds to Authenticate and display data
FPS Select Option:
1. Verify ID
2. Enroll ID
3. Delete ID
4. Quit
FPS VERIFICATION
FPS Press finger to start
FPS Captured
FPS Authentication PASSED with ID = 0
FPS Loop End reached
Motion detected, press Finger within 13100 seconds to Authenticate and display data
Current value = -19.531250 mA
Voltage value = -0.040283 V
Lux = 294.00
Thermocouple volt diff = 102.20 C
Temperature = 75.20 F , Humidity = 16.00
Sending Data to Server
S
AT+CIPSTART="TCP","184.106.153.149",80
R
80
CONNECT
OK
S
AT+CIPSEND=184
R
80
CONNECT
OK
S
GET https://api.thingspeak.com/update?key=JJAOBK32WOINKTOO&field1=0&field2=0&field3=-19.531250&field4=-0.040283&field5=294.000000&field6=102.198486&field7=75.199997&field8=16.000000
R
80
CONNECT
OK


Motion detected, press Finger within 12900 seconds to Authenticate and display data
FPS Select Option:
1. Verify ID
2. Enroll ID
3. Delete ID
4. Quit
FPS VERIFICATION
FPS Press finger to start
FPS Captured
FPS Authentication FAILED
FPS Loop End reached
Motion detected, press Finger within 12000 seconds to Authenticate and display data
FPS Select Option:
1. Verify ID
2. Enroll ID
3. Delete ID
4. Quit
FPS Loop End reached
Motion detected, press Finger within 5500 seconds to Authenticate and display data
Sending Data to Server
Motion detected, press Finger within 11900 seconds to Authenticate and display data
Sending Data to Server



Current, Voltage measurement using digital Multi-meter (For Verification)



Sensor value on Cloud server

4	2017-03-19 19:27:52 UTC	758	-22.216797	-0.040283	197
5	2017-03-19 19:28:10 UTC	759	-22.460938	-0.032227	197
6	2017-03-19 19:28:27 UTC	760	-22.460938	-0.040283	195
7	2017-03-19 19:28:45 UTC	761	-22.216797	-0.040283	199
8	2017-03-19 19:29:02 UTC	762	-22.460938	-0.032227	196
9	2017-03-19 19:29:19 UTC	763	-22.460938	-0.040283	198
10	2017-03-19 19:29:54 UTC	764	-22.460938	-0.040283	195
11	2017-03-19 19:30:29 UTC	765	-22.460938	-0.040283	196
12	2017-03-19 19:30:46 UTC	766	-21.972656	-0.032227	199
13	2017-03-19 19:31:05 UTC	767	-22.460938	-0.040283	198
14	2017-03-19 19:31:21 UTC	768	-22.460938	-0.040283	201
15	2017-03-19 19:31:39 UTC	769	-22.460938	-0.032227	199
16	2017-03-19 19:31:56 UTC	770	-22.460938	-0.032227	201
17	2017-03-19 19:32:14 UTC	771	-22.460938	-0.032227	201
18	2017-03-19 19:32:32 UTC	772	-22.460938	-0.032227	201
19	2017-03-19 19:32:48 UTC	773	-22.460938	-0.040283	201

Cloud Data Set of Sensor Value

CONCLUSION:

The FRDM K64f was used to collect data from the Ammeter, Voltmeter, Ambient light, Thermocouple, Temperature, Humidity, Motion Module and Fingerprint sensors, and this data was sent to the cloud using the ESP 8266 Wi-Fi chip and Thingspeak server.
The cloud data was verified with the actual measurements.

CHAPTER 6

DESIGN CHALLENGES AND SOLUTION

- Hexiwear Bootloader was in Base board.
    - Solution: Switching to FRDM K-64f with bootloader in the same board.
- Using 4 UART's from FRDM – 1 for Wifi Chip, 1 for Fingerprint, 1 for Serial debug and 1 for other clicks.
    - Solution: FRDM has 4 UART's total. We used 3, same UART for Fingerprint and Click BUS.
- Planning UART, SPI and I2C protocols and Analog pins without overlap.
    - Solution: Planning protocol pins as common for the bus, assign specific lines for SPI Chip select.

CHAPTER 7

FUTURE ENHANCEMENTS

- Using separate UART for Bus design.
- Using secure Wi-Fi chip.
- Improving Bus design with MUX for more devices.
- Reducing power consumption for devices.
- Designing enclosure for sensors.
- Edge programming to reduce data transfer load.
- AWS, Azure, IBM Watson integration.

CHAPTER 8

REFERENCES

- https://thingspeak.com
- https://www.mikroe.com/
- https://en.wikipedia.org/wiki/Hexaware_Technologies
- https://developer.mbed.org/compiler/#nav:/;
- https://developer.mbed.org/cookbook/Homepage
- http://www.4pcb.com/
- http://www.autodesk.com/products/eagle/