ECEN 5813 - PRINCIPLES OF EMBEDDED SOFTWARE

FINAL PROJECT PROPOSAL

Topic – Weather Monitoring Station

Name- Gauriech Ishaan Pilla

OBJECTIVE

As there have been many cases of fire in and around boulder, I came up with the idea of making a weather monitoring station. This project aims to continuously monitor climate using a temperature/humidity sensor (SHT21) to continuously monitor temperature and an MQ7 sensor that is suitable for sensing CO concentrations in the air.

FUNCTIONALITY OF THE PROJECT

The temperature sensor (SHT21) and the CO sensor (MQ7) will be utilized in this project, which is based on bare-metal programming. The I2C (Inter-Integrated Circuit) communication protocol will be used to connect SHT21 to the FRDM-KL25Z, while ADC will be used to connect MQ7 to the FRDM-KL25Z.

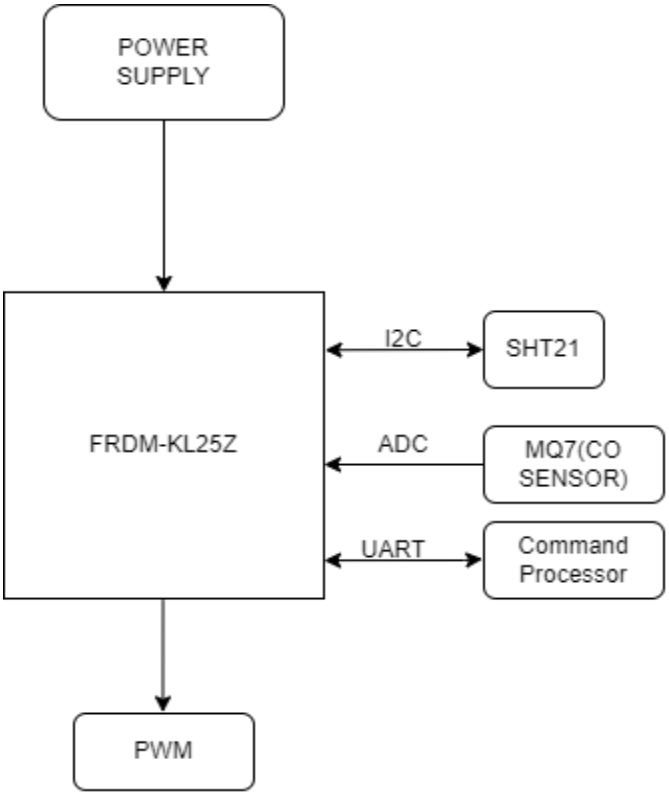
In this project, I will first read the data from both sensors. Then, using PWM (Pulse Width Modulation), I aim to glow the RGB LED according to the data read by the CO and temperature sensors, representing the state of the climate around this model. In addition, along with the data read by both sensors, an alarm message will be shown on the terminal window via the UART (Universal Asynchronous Receiver-Transmitter).

For Example – If the climate around the model is healthy, Green led would glow and according to the temperature and CO concentration, the LED would change from yellow to the final red state indicating the level can be harmful to humans.

TECHNOLOGY USED

I chose this project because it will let me to work extensively with I2C, ADC, UART, and PWM, particularly with I2C, which was taught in class but not included in any of the assignments. I wanted to delve deeper into the UART, ADC, and PWM concepts that I had learnt and implemented in earlier assignments. UART will be used to show messages on the terminal, PWM will be used to operate the RGB LED to signify various conditions, and ADC will be used to retrieve data from the MQ7 CO Sensor in my project.

BLOCK DIAGRAM



THINGS TO LEARN IN ORDER TO DEVELOP THE PROJECT

To interface the external temperature sensor with the FRDM-KL25Z board, this project requires the learning of an I2C communication protocol. I predict that learning the I2C protocol and how it works will be necessary before developing code for it. In addition, I must use ADC to interface an external CO sensor and display the data on UART. In assignment 7, I aim to learn about ADC, but I'll have to figure out extra procedures to get data from an external sensor.

Apart from that, because I had developed UART and PWM in the previous assignment, the implementation of these two concepts would strengthen the foundation of these two concepts. In addition, after my project's functionality has been established, I intend to implement a command processor.

REFERENCES

The textbook Embedded Systems Fundamentals with ARM Cortex-M based Microcontrollers: A Practical Approach by Alexander G. Dean covers the implementation and dealing with the UART, ADC, I2C, and PWM.

The NXP FRDM-KL25Z Reference Manual will be used to configure and work with the above-mentioned peripherals, as well as to learn about the architecture and use of various registers.

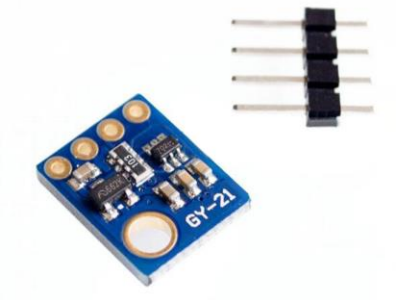
For interfacing sensors, I plan to read the datasheets for both sensors and figure out how to integrate them on my board using the NXP FRDM-KL25Z datasheet and reference manual.

HARDWARE USED

Two external sensors will be interfaced with the NXP FRDM-KL25Z Development Board that are:-

1. **SHT21**

   Link- https://www.amazon.com/HiLetgo-Digital-Humidity-Temperature-Replace/dp/B01N53H8SI/ref=sr_1_8?crid=3LJFB509SXVZ8&keywords=sht21&qid=1649803039&sprefix=sht21%2Caps%2C121&sr=8-8



This will be interfaced with the microcontroller over I2C protocol.

2. **MQ7**

   Link- https://www.amazon.com/ACEIRMC-Detector-Monitor-Arduino-Raspberry/dp/B0978KTWS3/ref=sr_1_2?crid=3UCCQ7ETKZX4&keywords=mq7&qid=1649803064&sprefix=mq7%2Caps%2C99&sr=8-2&th=1

This will be interfaced with the microcontroller using ADC.

TESTING STRATEGY

Since this project deals with both hardware and software, we will run a combination of automated and manual testing.

Automated Testing – Circular buffer testing is performed at the start of the program using the test cases from the previous Assignment.

Another automated testing that I would do is to make an array with some values between/including minimum and maximum values of my CO sensor and start a test at the start of my main function to show the different PWM values on RGB LED to demonstrate the working of CO sensor at different values. This code will run at startup if in DEBUG.

Note- I would like to thank Professor Howdy for sharing this idea for automated testing.

Manual Testing – The values read by the sensor can be tested if they are relevant to the actual environment i.e. whether the temperature and humidity sensor readings are accurate or not compared to the environment. Communication over I2C can be tested for packet transmission and reception using an oscilloscope or logic analyzer.