-: Fall 2021 :-

ECEN 5813

Principles of Embedded Software

Course Project Proposal

Digital Angle Gauge using Accelerometer

Author

Tanmay Mahendra Kothale

Tanmay.Kothale@colorado.edu

Objective: To implement the logic for a Digital Angle Gauge using MMA8451Q accelerometer.

Primary things to implement:

1. Figuring out the logic to interface MMA8451Q accelerometer with the FRDM-KL25Z board using I2C.
2. Implementing the on-board RGB LED to indicate the change of orientation of the board.
3. Changing the color of the RGB LED based on various events detected by accelerometer.
4. Implement an interactive command processor for debugging and calibration purposes.
5. Implementing Touch Sense Interface (TSI).
6. GPIO Interrupt to detect accelerometer tap event.
7. Circular buffers for UART communication.
8. State machine for command processing.

Additional hardware requirements:

There are no additional hardware requirements. If, in future, there is a need to add a hardware component, a proper documentation will be included in the readme file.

Technologies being used to implement the project:

1. Circular Buffer
2. I2C
3. Command Processing
4. State Machines
5. Interrupts
6. UART
7. Touch Sensor Interface
8. RGB LED

Any additional technology could be used in future to add new features to the project.

Functionality:

This project will be based on an interactive command shell which will allow user to type in the command and then do the operation as intended. This functionality is similar to what we did in assignment 6, but it also comes with additional points such as interfacing of Accelerometer, and the RGB LED and the TSI module. User will be able to type a certain set of commands over the UART and based on the command, the intended action will take place. Some of the commands include the functionality to calibrate the accelerometer, to change the LED color, to adjust the brightness of the LED. User can also adjust the brightness of the LED by using the TSI module. The RGB LED will display a specific color for each of the command executed. For example, if the user sends the command to calibrate the accelerometer, the LED will glow green. In a nutshell, a specific color will be allotted to each of the commands and whenever that command is being executed, the LED will be glowing with that color. If the system is waiting for the next command, the LED will be white, indicating that it is waiting for the next command. Whenever an invalid

command is entered, the LED will glow red indicating that a wrong command or a typo has occurred. I will also try to implement the arrow keys and tab completion functionality in my command prompt, if I have enough time to do so. The TSI sensor which is being used will be polled. User can change the desired angle of the accelerometer. When the accelerometer's desired angle is set, the LED will start blinking unless the desired angle is reached. After the desired angle is reached, the LED will stop blinking and display a solid color indicating that desired angle has been achieved.

In this project, the LED will act as nothing but an indicator for the user to determine whether the operation is being executed correctly or not. The TSI module will be used solely for the purpose of adjusting the brightness of the LED. I will also try to find out unique ideas to implement TSI in other functionalities as well, if time permits.

Testing Strategy:

1. Automated Tests
    a. Using automated tests to check the robustness of circular buffer
    b. Using automated tests to verify correct behavior of Accelerometer.
    c. Automated test sequence to check correct functioning of RGB LED's.

2. Manual Tests
    a. Manual tests to check whether the touch sensor is working properly.
    b. Testing of UART to check whether interrupt is being generated or not.
    c. Testing of UART implementation to check if the pipeline (using cbfifo) is working or not.
    d. Using interactive commands to control few operations such as changing LED color, calibration of accelerometer etc.

Things needed to learn:

1. I2C implementation to function correctly with accelerometer.

Sources:

1. Professor Howdy Pierce's lecture slides and reference codes.
2. Reference manual of KL25Z board.
3. Schematics of KL25Z board.
4. Reference codes of Prof. Alexander G. Dean.