Digital Angle Guage

Project Background

Back in the past, when a man has been cutting wood, he's also struggled to find an efficient and reliable way to cut precise angles. Thus a digital angle gauge can correct our saw to cut any precise angle. It can also be used for setting up a table by measuring the angle without a user needing it read from a scale. The core of our Digital Angle Guage is an accelerometer (MMA8451Q) which is inbuilt to our FRDMKL25Z board which should be calibrated to zero degrees before measuring any angle.

Functionality

1) Initially, the user will press a button connected to the FRDMKL25Z board need to calibrate it to zero degrees before measuring any angle and an LED is set up to indicate that calibration is completed and the user will also receive a message through a serial terminal.

2) The user then will provide the input of what angle he/she would like to measure with a digital gauge and input the angle via the user terminal

3) Once the input is fed to the terminal, the user will get the real-time status of whether the desired angle is reached through LED blinking at two different colors, RED - Desired angle not reached, GREEN - The desired angle reached and also through the user terminal

4) The user may wish to measure a different angle by providing the input to the command terminal without resetting the system.

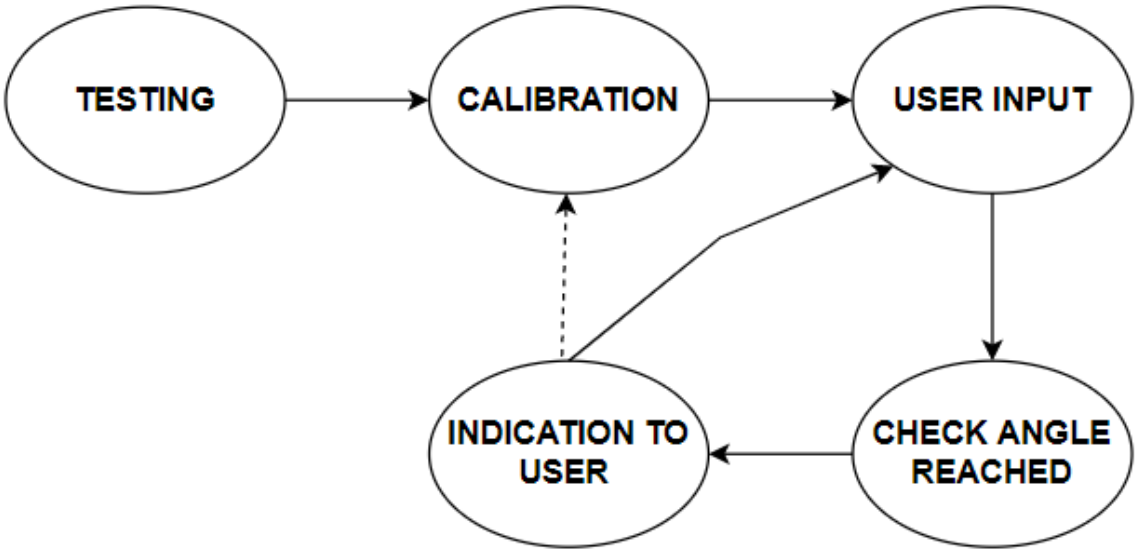

Fig 1: Functionality Flow

Technologies Used

1) UART - To create a user interactive terminal to provide input for digital angle gauge and to print real-time messages to the user

2) Circular Buffers - To achieve UART functionality

3) Command Processing - For accumulating user input, providing backspace capability when the user provides a wrong input, parsing user input, providing the help functionality to help the user to interact with the device(command processing may not be used extensively apart from the above functions). The command processing needs to be modified accordingly where the number which is in form of strings should be converted to integers before further processing.

4) Interrupt - To let the user calibrate the accelerometer for zero degrees, we need to ensure the critical section are handled properly, we are not bloating the interrupt handler with a calibration function and avoiding race around condition if we try to update a flag in multiple places

5) I2C - For interfacing the inbuilt accelerometer with FRDMKL25Z board

6) GPIO lines - To give visual feedback to the user whether the desired angle is reached through LED's. This will be event-driven rather than a blinking sequence as we used in our assignment 3

7) Statemachine - According to the functionality flow, the various states like Testing, calibration, getting input from users will be defined. Some cases when the user is trying to give input while we perform other operations like fetching data from accelerometer should be handled .

Additional Knowledge Required

1) Understanding how to interface accelerometer with FRDMKL25Z board through I2C protocol

2) Navigating through the datasheets to identify the registers that contain accelerometer raw values and getting reliable raw values by filtering out the anomalies.

3) Implementing calibration function for zero degrees offset during the initial setup

4) Converting the raw values into meaningful and accurate information (angle) to implement the digital gauge meter

5) Integration of the above-mentioned modules/technologies in a single project without affecting the functionality to ensure each module is properly encapsulated.
I would be referring to the following datasheets for reference
1) KL25Z Reference Manual
2) MMA8451Q Datasheet
3) Calibration basics
4) Accelerometer Calibration

Hardware Requirements:
All the necessary hardware required is available in freedom KL25Z freedom board. There is no additional hardware required except the switch which can be easily procured

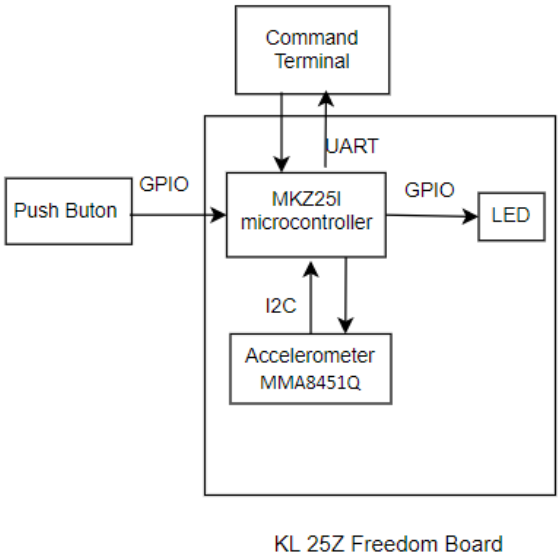

Fig 2: Block Diagram

Testing Strategy:

Automated Testing

1) The cbfifo implemented for UART functionality will be tested for multiple test cases
2) Test messages will be typed the user and echoed back to ensure the transmitter and receive functionalities are working fine
3) Interface between the accelerometer and FRDMKL25Z board is checked by reading basic registers to ensure proper communication.
4) The LED's will be checked through a blinking sequence

Manual Testing

1) To check whether the accelerometer is working, the board can be tilted left/right and the corresponding messages can be displayed in the terminal
2) The LED can be changed to a different color when the user presses the GPIO button to ensure the interrupt is triggered
3) Testing the behavior when a user gives an invalid command/angle whether it gives the right debug message
4) To check whether the desired angle is reached by manually measuring it for different sets of angles and testing it multiple times to ensure minimum deviation.