ELECTRICAL AND ELECTRONICS ENGINEERING DEPARTMENT

Laboratory Project Temperature-Initiated Object Detection

Laboratory Project - Temparature-Initiated Object Detection

Objectives

In the EE447 laboratory work, you were expected to familiarize yourself with the operation of TM4C123G and its utility modules. In this final project you are expected to gather the previous experience on the microcontroller with novel information to achieve a multi-functional task. The objectives of the project are as follows:

- Interpretation of the necessities of a complex task and separation into sub-task
- Fulfillment of co-operation of utility modules
- Understanding a given complex hardware and compatibility of its components
- Writing a multi-task software for a given complex setup
- Introduction to the serial communication on TM4C123G and utilization of the protocols I^2C and SPI.

1 Project Definition

In this project, you are expected to build a temperature-initiated object detector based on two different temperature sensors and an ultrasonic distance sensor. After making all necessary initializations, your system should go into deep sleep mode. While in deep sleep mode, the system will be able to detect a possible temperature increase using an LM35 analog temperature sensor. If the temperature of this sensor rises above a threshold, the system will wake up from deep sleep and take a measurement from the higher precision BMP280 temperature sensor through I2C. If it is confirmed that the temperature is above the threshold, the system starts scanning the environment for close by objects in the environment. For object detection, an ultrasonic sensor should be used through a GPTM module. The sensor is mounted to a stepper motor's shaft, and can scan an angle interval of -90 to 90 degrees. The system takes distance measurements within this angle interval, and if it detects an object, shows its distance and angle relative to the system. To show the current configuration and measurements, you will use a Nokia 5110 LCD display, which you will drive using the SPI module. You will also use GPIOs for the on-board RGB LED, keypad and pushbuttons, an external power LED, and GPTM for an external speaker.

Overall, the system has three major functions:

1.1 Heat Sensing in Deep Sleep Mode

In deep sleep, the system keeps tracking the temperature using an analog temperature sensor, LM35. The analog temperature threshold is adjustable via a multiturn trimpot. To detect the temperature rise above the given threshold, the analog comparator peripheral of the TM4C123 has to be used. If the temperature measured with LM35 is above the threshold, a temperature measurement is taken from BMP280 digital sensor.

Note: In the test, we will simulate an increased temperature by a lighter.

1.2 Object Detection

If the temperature measured from the digital sensor is above the threshold, the system scans an angle interval of -90 to 90 degrees and takes distance measurements to detect an object. To scan the angle interval, an ultrasonic distance sensor is mounted on a stepper motor. You can assume that the motor has fixed step angles; that is, if you apply a fixed number of steps the motor turns by a fixed angle. You can also assume that the object's distance is at most 1 m. That is, if the distance measurement is larger than 1 m, then no object is detected at that angle.

1.3 User Interface

The system has four output elements.

1. The temperature detected by the digital temperature sensor, the temperature threshold, and the information if an object is detected together with its angle and distance are displayed on the LCD screen.
2. If the analog temperature sensor detects an increased temperature level, a power LED is turned on.
3. If the digital temperature sensor detects heat above the threshold, a speaker plays a sound
4. If an object is detected, according to its distance, the onboard RGB LEDs are driven.

Moreover, the system has three input elements

1. A multiturn trimpot is used for analog temperature threshold adjustment
2. A 4x4 keypad module is used for digital temperature threshold adjustment.
3. A button is used to enter deep sleep after all operations are complete.

2 Requirements and Restrictions

The overall functionalities of thermal object detection system are described in Section 1. For the sake of simplicity there will be some assumptions about itsoperation. Additionally, there will be restrictions on both the implementation and the hardware to be used.

2.1 Requirements

You will use LM35 and BMP280 temperature sensor and a HC-SR04 ultrasonic distance sensor as input elements and a stepper motor as output element. You will also use a Nokia 5110 LCD display, a power LED, a speaker, onboard RGB LEDs, a potentiometer, a 4x4 keypad, and the onboard pushbutton as the user interface. Functional requirements are as follows.

1. At deep sleep, only the digital temperature threshold should be shown on the LCD. Other data, such as temperature measurement, detected object's angle and distance should be cleared before entering deep sleep. Power LED, speaker and RGB LEDs should be off.
2. When the analog temperature sensor detects a temperature above the threshold and the system wakes up, the power LED should be turned on. It should remain on until the system enters deep sleep mode again via a button push.
3. When the digital temperature sensor detects a temperature above the threshold, the speaker should play a square wave for 2 seconds, and then turn off.
   BONUS: The pitch (frequency) of the sound can change according to the detected temperature.
4. After the digital temperature detects heat increase above the thershold, the stepper motor - distance sensor system should scan a range of -90 to 90 degrees. This scan should be done only once. If no object is detected via the distance sensor, a message such as"NO OBJECT" should be displayed on the LCD. RGB LEDs should be off. If an object is detected, its angle and distance should be displayed.
   (a) If the detected object's distance is between 75 cm and 100 cm, onboard green LED should be turned on. Red and blue LEDs should be off.
   (b) If the detected object's distance is between 50 cm and 75 cm, onboard blue LED should be turned on. Red and green LEDs should be off.
   (c) If the detected object's distance is less than 50 cm, onboard red LED should be turned on. Green and blue LEDs should be off.

   BONUS: You can show an angle vs distance plot similar to Fig. 1.
5. The system should have a button to enter deep sleep mode.
6. **BONUS:** The system can also show the the temperature measured by LM35 and threshold set by the trimpot. Note that analog comparator pins are not ADC inputs. Thus, you may need to connect these components to multiple pins.

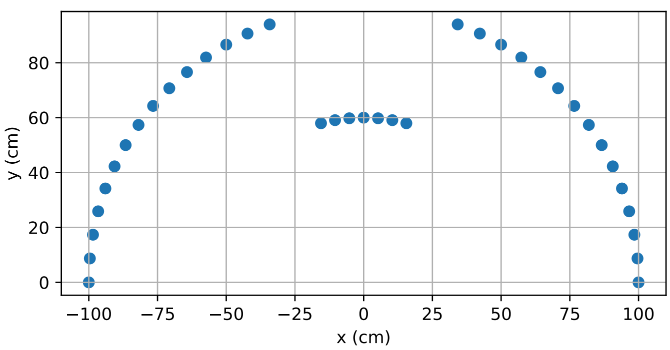

Figure 1: Distance - angle plot example (BONUS)

2.2 Restrictions

2.2.1 Hardware Restrictions

The project uses the following components:

1. NOKIA 5110 LCD Screen
2. Step Motor
3. HC-SR04 Ultrasonic Distance Sensor
4. LM35 Analog Temperature Sensor
5. BMP280 Pressure and Temperature Sensor
6. Speaker
7. Trimpot
8. 4x4 Keypad
9. Transistors
10. 1W Power LED

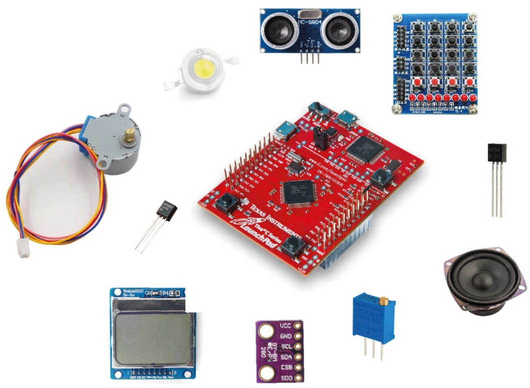

Figure 2: Components

2.2.2 Implementation Restrictions

To ensure the learning outcomes of the course and this laboratory, there are a few restrictions on implementation:

- LM35 Temperature Sensor
  1. LM35 must be connected to the analog comparator module.
  2. Analog comparator interrupts must be configured to wake the system from deep sleep.
- BMP280 Temperature Sensor
  1. BMP280 must be read using the I2C module.
  2. As a simple lowpass filter, 128 temperature readings must be collected and their average must be taken to calculate the detected temperature.
- Power LED and buzzer can draw more current than a GPIO pin can supply. To prevent damage, they must be driven using transistors. They must be powered from an external power supply.

The visual style of the user interface on Nokia LCD screen is up to you as long as you fulfill the requirements specified in Section 2.1. To be eligible for bonus points from the project, you have to fulfill the requirements described in Section 2.1 and satisfy the restrictions described in Section 2.2.

2.2.3 Programming Language Restrictions

- You need to implement at least one submodule or function in ARM Assembly.
- You can use C programming language for the remaining sections.

3 Tips

You will need to use interrupts and configure the priority of the interrupts. Recall that to configure an interrupt at register level, you should know the interrupt number of the interrupt source you plan to use so that you can decide which NVIC registers (see page 141 of [1]) to be configured. You can find the interrupt number of an interrupt source from the table in page 104 of [1]. Alternatively, you can use `NVIC_EnableIRQ` and `NVIC_SetPriority` functions.

You can read BMP280 using I2C interrupt handler and the ultrasonic distance sensor using a GPTM interrupt handler. Thus, you may do other repetitive tasks in an endless loop inside main. An example algorithm for the main loop is given below.

1. Wait for I2C handler to store 128 readings in an array.
2. Calculate the temperature as their average.
3. Drive the speaker, if necessary
4. Scan an angle range between -90 and 90 degrees. Take distance measurements along the rotation.
5. Detect an object at the angle with minimum distance measurement.
6. If the Sleep button is pressed, enter deep sleep. After the system wakes up, loop back to 1.
7. If more than a second has elapsed after the last LCD update, take a new temperature measurement from BMP280. Update the temperature value on LCD
8. Loop back to 6.

You can use standard C libraries, such as `stdio.h`, `string.h` and `math.h`. Moreover, only for driving Nokia 5110 LCD, you can use Prof. Valvano's functions [2]. **Any other third party code usage is prohibited.**

For detailed information about analog comparator see Section 19 of [1].

For detailed information about deep sleep mode see Section 5.2.6.3 of [1].

For detailed information about I2C module in TM4C123 see Section 16 of [1].

4 Deliverables

You are supposed to attempt the project work in group of two.

- **Q & A Session:** There will be a question/answer session in the last week of December. You can focus on asking about the usage of the deep sleep mode and the LCD. **Date: will be announced**
- **Final Report:** A full description of your work, including photos of your setup and relevant portions of code you may need to explain certain key points. Full source codes are not required in your report. Please clearly indicate how you fulfill the requirements and satisfy the restrictions. Moreover, your codes should be well-commented. **Date: 15.01.2024**
- **Final Video** A video that demonstrates the working project. At most 5 minutes. **Date: 16.01.2024**
- **Source Code:** A zipped Keil μvision project folder with fully executable codes is to be uploaded to ODTUClass. **Date: 14.01.2024, 12:00**
- **Lab Demo:** Demonstration of the operation of the project. **Date: 15.01.2024**

References

[1] TI, "Tiva™ tm4c123gh6pm microcontroller data sheet." http://www.ti.com/lit/ds/spms376e/spms376e.pdf.

[2] J. Valvano, "Starter files for embedded systems." https://users.ece.utexas.edu/%7Evalvano/arm/.