LOW POWER EMBEDDED DESIGN PROJECT UPDATE #1

Team Name : WearTech

Team Mates:

Sanjana Kalyanappagol
saka2821@colorado.edu

Shekhar Satyanarayana
shsa5563@colorado.edu

Components Required:

- Processor: Blue gecko BGM11S module
- As discussed, our project requires 2 Blue Gecko dev kits to act as devices in the mesh network.
- Sensor: Triple Axis Accelerometer ADXL345
- Sensor: Capacitive sense PCF8885

Part Selection:

Processor: Blue gecko BGM11S module

We chose this processor because it supports Bluetooth mesh networking and has higher range for communication.
https://www.silabs.com/products/wireless/bluetooth/bluetooth-low-energy-modules/bgm11s-bluetooth-sip-module

Sensor: Triple Axis Accelerometer ADXL345

We chose this sensor because accelerometer plays an important role in the project to identify characters / command given by the user. This sensor could be used to obtain data according to each axis readings and then process the data to identify the character.
Serial port interface: I2C. Blue Gecko has two I2C interfaces available.
Power supply: This sensor requires regulated power supply.

Features and Benefits:

- Ultralow power: as low as 40 µA in measurement mode and 0.1 µA in standby mode at VS = 2.5 V (typical)
- Power consumption scales automatically with bandwidth
- User-selectable resolution
- Fixed 10-bit resolution
- Full resolution, where resolution increases with g range, up to 13-bit resolution at ±16 g (maintaining 4 mg/LSB scale factor in all g ranges)
- Embedded, patent pending FIFO technology minimizes host processor load
- Tap/double tap detection
- Activity/inactivity monitoring
- Free-fall detection
- Supply voltage range: 2.0 V to 3.6 V
- I/O voltage range: 1.7 V to VS
- SPI (3- and 4-wire) and I2C digital interfaces
- Flexible interrupt modes mappable to either interrupt pin
- Measurement ranges selectable via serial command
- Bandwidth selectable via serial command
- Wide temperature range (−40°C to +85°C)
- 10,000 g shock survival
- Pb free/RoHS compliant
- Small and thin: 3 mm × 5 mm × 1 mm LGA package

Why this sensor? Why not MMA8542Q?

This sensor is a ultralow power sensor. It’s stand-by mode current is 0.1uA and 40uA in measurement mode which is less as compared to MMA8542Q sensor.
Load power management can be performed as its supports different energy modes.In Blue gecko, the I2C can perform address recognition in EM3-Stop mode.

It is an interrupt driven sensor so the MCU is turned on and processes the data only when we receive data via I2C on a GPIO pin.

Sensor: Capacitive sense PCF8885

Why this sensor?
Capacitive 8-channel touch and proximity sensor with auto-calibration and very low power consumption. The integrated circuit PCF8885 is a capacitive 8-channel touch and proximity sensor that uses a patented (EDISEN) method to detect a change in capacitance on remote sensing plates. Changes in the static capacitances (as opposed to dynamic capacitance changes) are automatically compensated using continuous auto-calibration. Remote sensing plates (for example, conductive foils) can be connected to the IC1 using coaxial cable. The eight input channels operate independently of each other. There is also a built-in option for a matrix arrangement of the sensors: interrupt generation only when two channels are activated simultaneously, suppression of additional channel outputs when two channels are already active.

The Capacitive sensor is interrupt driven. This sensor uses regulated power supply.

Why Not SparkFun - AT42QT1010 for Capacitive sense?

1. Even though the AT42QT1010 is low power capacitive sense , it does not use I2C communication. Even the current is more compared to the PCF8885

Features and Benefits:

- Dynamic touch and proximity sensor with 8 sensor channels
- Support for matrix arrangement of sensors
- Sensing plates can be connected remotely
- Adjustable response time
- Adjustable sensitivity
- Continuous auto-calibration
- Digital processing method
- Can cope with up to 6 mm of acrylic glass
- Direct and latching switch modes
- I 2C Fast-mode Plus (Fm+) compatible interface
- Two I2C-bus addresses
- Cascading of two ICs possible
- Interrupt signalling over I2C-bus
- Large voltage operating range (VDD = 2.5 V to 5.5 V)
- Sleep mode (IDD < 100 nA)
- Low-power battery operation possible (IDD ~ 10 A)
- Operating temperature range (Tamb = 40 C to +85 C)
- Available in TSSOP28 and SOIC28 package

Radio Communication:
Chosen communication : Bluetooth Low Energy, Bluetooth and Bluetooth Mesh Network.
Why Bluetooth Technology?

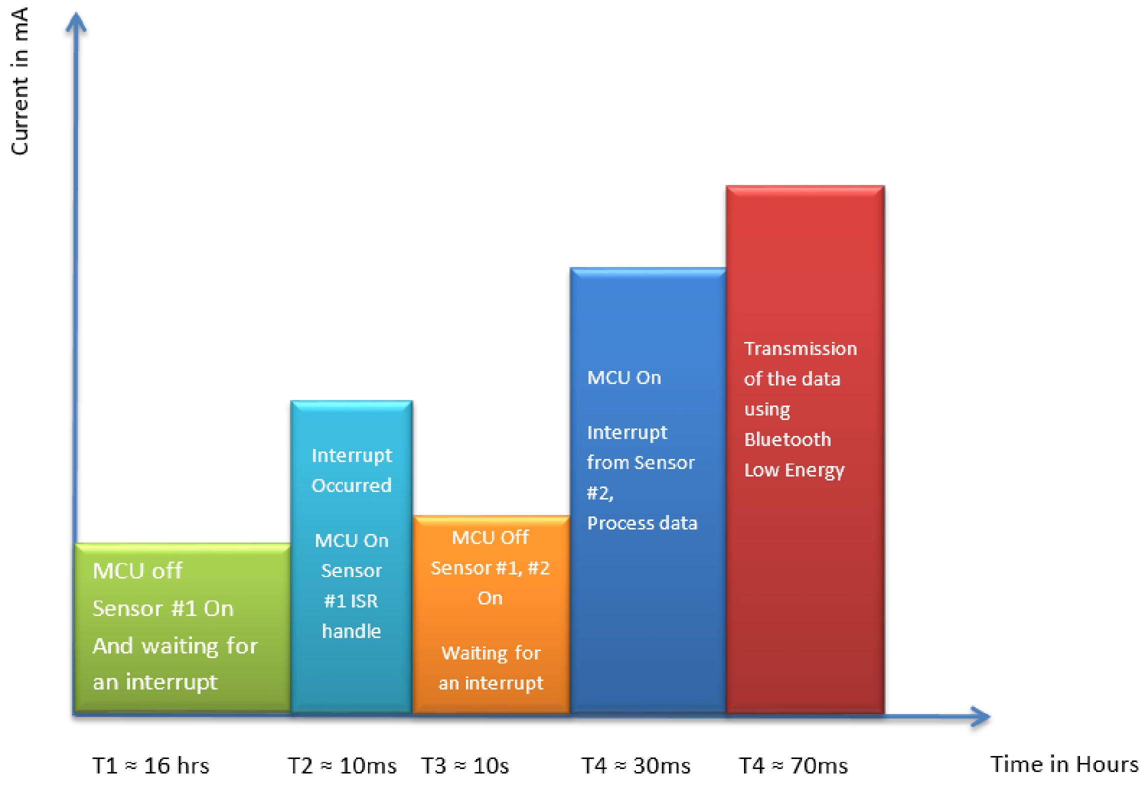
1. Bluetooth technology is most secure and is resistant to other radio communications like wifi/zigbee and even ANT. Since it has more resistance to interference from other technologies, we opted BLE.

2. Since Bluetooth Mesh is a new technology, this course and project provide us an opportunity to explore it. Thus not exploring thread/zigbee mesh networks.
3. With Wifi, we would be using a lot of power as compared to BLE. Our project does not need a higher bandwidth communication technology, thus wifi is not a good option.

Use Case Model:

Use case model 1:

(In terms of a single day usage from the customer/user)
Considering a person who sleeps for 8hrs, making a total of 16hrs of the day in which our device should be in on-state.
With a generic user, if we presume that the device would be used for almost 8hrs in a day.
Assuming an ideal user, who would switch on the device and start using it/ does no actions for 10seconds after switching on the device.



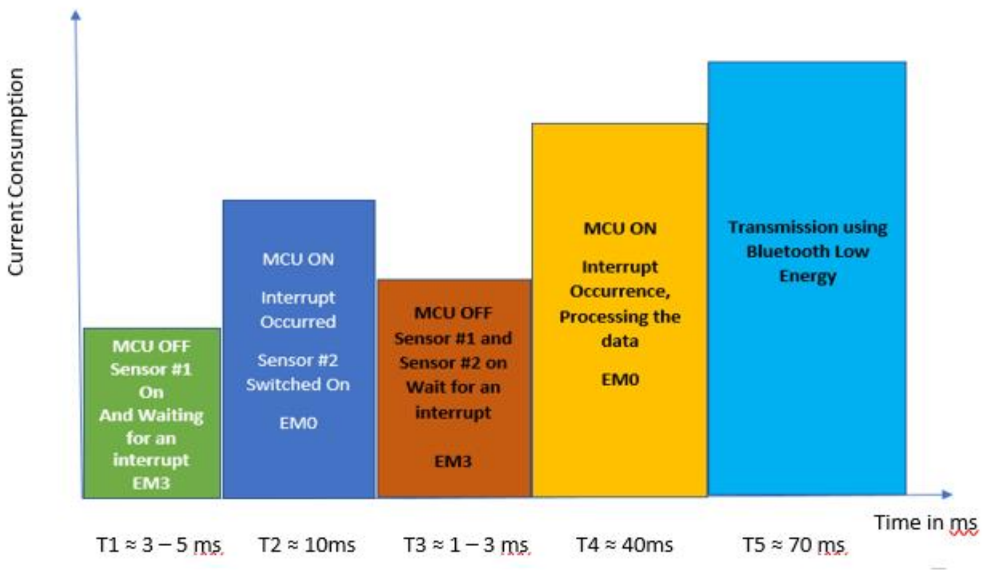
T1: The MCU can wait for an interrupt for a stretch of 16 hrs
T2 : Once Interrupt of Sensor #1 is occurred the MCU is on for 10ms to process the information and turn on the sensor #2
T3 : Once the sensor #2 is switched on, the MCU would be in sleep mode waiting for interrupts from Sensor #1 & Sensor#2. If we have received no interrupt from Sensor#2 after say 10seconds, we will

disable the sensor#2.
[The User might switch on the device and never do any action till the next 10seconds, thus we are making sure that the accelerometer is turned off if no actions are captured from sensor#2 after 10seconds]
T4: Once we have received the data from the user, the interrupt is generated and we process the data. This approximately takes around 30ms to process the data.
T5: The processed data has to be sent to the mobile Application within the approximate 70ms.

Note:

1. For this use case the worst case time is considered.
2. After the data is transmitted through BLE, the mobile device should accept the connection and start processing the data. And take according actions/ record the actions. All these has to be computed and accomplished in realtime, thus should take as less time as possible, say within a second.

Use Case Model 2:



Use Case model condition: When the user switches on the device and immediately sends the data (performs an action)

State 1: MCU is off, Sensor #1 i.e Capacitive sense is On and it is waiting for an interrupt to switch on the entire device. The device stays in this state for about 3-10 ms after the user has used touch sense to switch on the device. The capacitive sense we have chosen uses I2Cs serial port interface which works in EM3 for Blue Gecko and also, we wait for an interrupt in EM3 energy mode.

State 2: MCU becomes on when an interrupt occurs, hence it will be in EM0 mode. In the interrupt service routine sensor #2(Accelerometer) is switched on. This state will take about 10 ms to handle the interrupt and switch on sensor #2.

State 3: In this state, MCU is off while sensor #1 and sensor #2 are on. Sensor #2 is waiting for an interrupt i.e sensor #2 is waiting to receive data from the user. As use case considered is when user switches on the device and immediately writes the data, this state will approximately take 1-3 ms.

State 4: In this state, MCU becomes on when an interrupt occurs i.e when data is received from the user and the data is processed. This state will approximately take 40 ms to handle the interrupt and process the data.

State 5: In this state, the data is transmitted to the mobile phone using BLE and it is interpreted to display the character received. This will take about 70 ms.