# ECSE426 Microprocessor Systems
# Winter 2018
## Final Project
# IoT – Sensor Data Management from Hardware to Cloud

## Introduction

For the final project, we will develop a system to explore the interaction of embedded peripherals and sensors with cloud-enabled services, which is one of the main hallmarks of Internet of Things (IoT) designs. The F4-Discovery board will be augmented by a Bluetooth Low Energy (BLE) connection to connect to Internet via a smartphone. The BLE interface will be realized through a STM32F401RE Nucleo board, along with an IDB04A1 BLE daughter board, which will be connected to the Discovery board by a serial link. The BLE module should transmit and receive data between the F4 Discovery and the smartphone. The smartphone will employ the cloud services for upload, download and processing of the files.

The system will allow the board to send audio and accelerometer data and other optional data such as a push button status, over the BLE connection to the smartphone device. This data will be saved in a file and uploaded to the FireBase cloud services. The processing such as filtering can be done on chip or online in the cloud service. Students are free to choose any type of processing or filtering with proof of advantages.

## Project Details and Design

The project is composed of four units. Each of the units needs to accomplish specific function. The code should be written using freeRTOS or RTX operating systems. Functionality Outline by Hardware Parts:

1. STM Discovery board
    - Read the audio (Electret Mic Breakout) data (by single tap detection),
    - Or, Read the accelerometer data for 10 sec (by double tap detection),
    - Apply processing if needed,
    - Transmit data serially (UART) to the Nucleo board.
2. STM Nucleo board with BLE board
    - Interconnects Discovery board with smartphone. Provides BLE functionality to the Discovery board.
    - This serial communication should be done via a UART connection, at the fastest Baudrate you can achieve.
3. Smartphone
    - Interconnects Nucleo board with cloud service. Access FireBase cloud services, including the authentication and processing the data. Sends processed data from the cloud back to Nucleo and then Discovery board if needed.
4. Cloud Service
    - Stores Discovery board data, manipulated files, makes data visible to clients on any platform, and apply basic processing such as filtering.

- Performs advanced processing such as voice recognition and decision. Detect the recorded voice and return a number based on the detected voice.

## STM Discovery Board

The purpose of the Discovery board is to read the accelerometer data to detect single or double tap detection:

- If one tap is detected, record audio data for one second. The voice should be one to two digits number. Optionally, this data could be a simple “Yes”, “No”, “Go” or any simple one-word sound. Then, one LED should be turned ON after a short delay indicating the microphone is recording. The data should be sampled at speeds higher than 8K Samples/sec, higher speeds results higher sound quality. Once the record is done, the data should be transmitted to the Nucleo board. The Tx/Rx duration should be represented by another LED to let the user know the start and end of data transfer. Once the sound is processed and the feedback is sent back to the discovery board (expecting a number N), the discovery board should turn ON and OFF one LED for the N times.

  For example: The user taps once on the board and after about a second an LED turns ON indicating the user should say something. If the user says “Four”, the recorded sound should be transmitted to the Nucleo, then to your phone and finally to the cloud service. The signal should be processed to reveal the detected number. Therefore, a number “4” will be returned back to Nucleo and then Discovery board. Now, the Discovery board should toggle an LED 4 times.

- If two taps are detected, transmits the pitch and roll values for 10 seconds sampled at 100 Hz. Plot its graph in your cloud service.

## BLE Transceiver (STM32F401RE Nucleo board + IDB04A1 BLE board)

The purpose of the Nucleo board is to provide the Bluetooth Low Energy functionality to F4-Discovery board. Using a BLE daughter board, the Nucleo board will connect the Discovery board to the outside world where it can eventually reach the cloud. The Nucleo board needs to obtain the data readings from the Discovery board and transmit them to the phone over BLE, and vice versa. The drivers and API for using BLE are provided, and the documentation can be found in (DOC_20 to DOC_23).

There will be at minimum a microphone and a tilt data BLE services. Their characteristics will have properties ‘READ’ and ‘NOTIFY’. The button service (optional) will need the property ‘NOTIFY’.

## Android/iOS application

### BLE

You will have to connect your phone to the board using Bluetooth Low Energy and obtain the readings from it and store them to the cloud. It is recommended to use an Android smartphone, and a good IDE to use is Android Studio. If you wish to proceed with iOS devices is optional, or if you need assistance acquiring an Android device, do not hesitate to contact the course Instructor.

For this section, reference material will be provided, and it will be covered during the tutorial. For a good example of code, the Android developer website contains a sample application of BLE use. This application searches for all BLE devices, and after connecting to a device provides all services and characteristics provided by the device. This application can only receive the data from BLE.

## Cloud services

A short tutorial for the FireBase cloud services will be provided by a TA.

# Demonstration

There will be two stages of demonstration

- **Progress Demo** will take place during the week of **April 3 to 6**. Every group is obliged to provide the specification of all functionality that will be provided during the final demo. Exact dates will be announced.
- **Final demo** will take place on **Friday, April 13 (being announced later).**

# Report

The final project report is supposed to be more formal than the lab reports which you wrote during this semester. You should naturally consider all feedback you have received this semester while preparing your report. This report should be like **a user manual** for a new person whose using your system. In particular, we would like to stress the need for the extra following points in your report:

- The used components in your system.
- A timeline of work and a breakdown between team members.
- A block diagram of your system, showing roughly how modules interact.
- Proper screenshots of your application.

As always, you should explain the reasoning behind your design choices and provide sufficient information on how the details are realized. The contributions of each member should be clearly delineated in one place in a concise manner. The working code on embedded, phone, and cloud sides will need to be included in the project submission, in **one zip file**. The code by itself should be clean and well-documented. Please try to reduce the size of submission file.

# Notes:

1. If you plan to use CubeMx to create your base project, you need to add the "LIS3DSH.c" and "LIS3DSH.h" files to your project. You need to enable the SPI1 as well as other peripherals you need to use. The accelerometer sensor communicates with uP through the SPI protocol.
2. As a starting point, begin with the sample code for Discovery and Nucleo boards. For the Nucleo, compile and download the code to the board. Install a software like BLE scanner, connect to the "BlueNRG" and open the first service. Tap on **R** (near **N**) on the second characteristic to request a read. You should get all **0** as a value. Now, press the **blue** button on the board and then tap on the same **R** again. You will see the value changes every time you press the **blue** button. This is an increasing variable if you locate the code for the Nucleo board. If you tap on **N**, you are saying the properties is "Read and Notify". So, every time you press blue button, the value on your phone gets updated automatically. Locate the UUIDs and match these numbers with the base project code of Nucleo board.

# Important! Obtaining and returning your Kits

Final project group kits, Nucleo and microphone, are ready to pickup once announced. All groups should return all components to ECE labs within one week of the final project demo. The parts returned should exactly match the ones given to you (specs, models, and part numbers). They should all be in a fully working condition. This include all boxes, kits (discovery and wireless), tools (screwdrivers and wire wrappers), breadboards and peripheral components. On the occasion of your failure to return the kit on time, or having any missing components, you will be penalized 40% of the total project grade.