

Sneak Review

**Sneak Review of in session workshop materials,
with full materials delivered in class atmosphere**

Tutorials: Embedded Linux Conference 2017



Researching Technologies

Configure Arduino WiFi: Manipulate LEDs without programming.

Description: Follow online Arduino instructions (<http://www.arduino.org/learning/getting-started/getting-started-with-arduino-uno-wifi/>), and then test (1) connectivity and (2) the Arduino WiFi's preprogrammed RestServer application by typing in a computer web browser:

```
http://192.168.240.1/arduino/digital/13/1
```

...to turn on the Arduino WiFi's D13 LED. In case the IP address is not 192.168.240.1, then use the correct one.

Configure Arduino Lucky: Program a Arduino Lucky Shield application.

Description: Examine the online Lucky Shield reference guide (<http://www.arduino.org/learning/reference/lucky-begin/>), and program a connected Arduino board accordingly.

Develop TCP/IP on Arduino WiFi: Program a Web server application.

Description: Examine the online web server tutorial (<http://www.arduino.org/learning/tutorials/boards-tutorials/webserverblink/>) and implement a web connected LED button.

Tip: Other sample projects for this platform appear online (<http://www.arduino.org/products/boards/arduino-uno-wifi/>).

Extra: For extra credit, implement a REST web service (<http://www.arduino.org/learning/tutorials/boards-tutorials/restserver-and-restclient/>) instead.

Configure NXP FRDM-KL25Z: Program an introductory ARM mbed application.

Description: Follow online ARM mbed instructions (https://docs.mbed.com/docs/mbed-os-handbook/en/latest/getting_started/blinky_compiler/), and then observe the blinking LED1 (red) on the FRDM-KL25Z microcontroller.

Extra: Extend the application with color and frequency concepts suggested in documented challenges (https://developer.mbed.org/teams/mbed/code/mbed_blinky/).

Work with the NXP FRDM-KL25Z: Program more rich ARM mbed applications.

Description: Follow online ARM mbed examples (<https://developer.mbed.org/handbook/mbed-FRDM-KL25Z-Examples/>) to become familiar with the rich sensor and actuator features of this microcontroller.

Configure Nordic Semiconductor nRF51-DK: Test onboard LEDs.

Description: Get started in the usual way, writing or importing simple applications in ARM mbed and programming a connected nRF51-DK board. To test all LEDs on this board, consider the following light chasing logic:

```
#include "mbed.h"

DigitalOut Boardleds[] = {(LED1), (LED2), (LED3), (LED4)};

int main() {
    int nIter;
    int Previous;

    while (1) { // Loop forever
        for (nIter = 0; nIter < 4; nIter++) {
            if (nIter == 0)
                Previous = 3;
            else
                Previous = nIter - 1;

            Boardleds[nIter] = 1;
            Boardleds[Previous] = 0;

            // Wait(0.2 secs);
            Thread::wait(500);
        }
    }
}
```

Develop Bluetooth Smart on nRF-51DK: Program a Physical Web appliance.

Description: Examine the online Physical Web project guide (<https://developer.mbed.org/teams/mbed-os-examples/code/mbed-os-example-ble-EddystoneService/>), and observe a new Eddystone beacon appearing in the Physical Web smartphone application.

Tip: Install a mobile (smartphone) app to test the nRF-51DK appliance.

Explore H2M Bluetooth Communication: Install Bluetooth smartphone apps.

Description: A wealth of possibilities exist to communicate on mobile devices with human interfaces. To explore H2M Bluetooth communication, complement existing applications running on nRF-51DK boards with mobile applications running on telephones:

Android nRF Blinky (<https://play.google.com/store/apps/details?id=no.nordicsemi.android.nrfblinky>) Physical Web (https://play.google.com/store/apps/details?id=physical_web.org.physicalweb) nRF Connect (<https://play.google.com/store/apps/details?id=no.nordicsemi.android.mcp>)

iOS Physical Web (<https://itunes.apple.com/us/app/physical-web/id927653608/>) nRF Connect (<https://itunes.apple.com/us/app/nrf-connect/id1054362403/>)

Windows Bluetooth Beacon Interactor (<https://www.microsoft.com/store/p/bluetooth-beacon-interactor/9nblggh1z24k/>)

Any OS (Operating System) nRF Blinky (<https://nordicsemiconductor.github.io/nrf5-physical-web/projects/nRF-Blinky/>)

Install MQTT client: Install applications to support MQTT without programming.

Description: Issue the package installation command (according to OS (Operating System) platform) to install **mosquitto-clients**. This step is required before carrying out commands with *mosquitto_sub(1)* and *mosquitto_pub(1)*.

Instructor's reminder: A cost free account on CloudMQTT (<https://www.cloudmqtt.com/>) facilitates this tutorial.

Configure Arduino MQTT: Enable basic MQTT client reporting.

Description: As shown in the web panel (<http://www.arduino.org/images/tutorials/connectivity.png>), the board reports its status to a MQTT broker if configured to do so. Select 'Enable MQTT client' and ensure that 'MQTT client state' reports 'connected.'

Developing Appliances

Build a Weather Station Appliance: Monitor vitals and calculate dew point.

Description: Device: Arduino WiFi.

Build a Volume Control Appliance: Monitor gesture and finger position.

Description: Device: FRDM-KL25Z.

Build a Motion Control Appliance: Monitor joystick activity for robotics.

Description: Device: Arduino WiFi.

Build a Air Mouse Appliance: Monitor hand gestures and move a mouse cursor.

Description: Device: FRDM-KL25Z. USBDevice (https://developer.mbed.org/users/mbed_official/code/USBDevice/) is a ARM mbed library that implements the USB human interface device (HID) class, among other USB specification parts. The USBDevice library is instrumental in building HID appliances like the air mouse, along with the MMA8451Q (<https://developer.mbed.org/users/emilmont/code/MMA8451Q/>) library which provides accelerometer data.

Tip: The NXP MMA8451Q circuit is a semiconductor that reports accelerometer data over a I²C serial interface, a communication standard invented by NXP.

Build a Heart Rate Appliance: Monitor heart rate over a Bluetooth connection.

Description: Device: nRF51-DK. Examine the online Heart Rate project guide (<https://developer.mbed.org/teams/mbed-os-examples/code/mbed-os-example-ble-HeartRate/>), and observe a new HeartRate device appearing in the nRF Connect smartphone application. The source code uses simulation logic to remedy the assumed absence of a real sensor.

Tip: Install a mobile (smartphone) app to test the nRF-51DK appliance used in development.

Build a Crash Detector Appliance: Monitor vibrations matching crash patterns.

Description: Device: FRDM-KL25Z, detect (directional) tap sequences using the onboard MMA8451Q sensor and indicate vehicle crashes.

Tip 1: The secret knock project (https://developer.mbed.org/users/maclobdell/code/FRDM-KL25Z_secret_knock/) can be used as a template to build new vibration detection logic.

Tip 2: A NXP document (<https://community.nxp.com/docs/DOC-329888/>) helps understand how the MMA8451Q sensor works.

Build a Intrusion Detector Appliance: Implement access controls and alarms.

Description: Device: Arduino WiFi, detect and indicate motion with the shield mounted PIR sensor and buzzer actuator.

Build a Light Switch Appliance: Monitor presses and remotely switch a lamp.

Description: Device: nRF51-DK, simulate a lamp with onboard LEDs and control with an H2M mobile application.

Examine the online LED control project guide (<https://developer.mbed.org/teams/mbed-os-examples/code/mbed-os-example-ble-LED/>), and manipulate an onboard LED represented as an icon appearing in the *nRF Blinky* smartphone application.

Tip: *Install a mobile (smartphone) app to test the nRF-51DK appliance.*

🚩 Build a Street Lamp Appliance: Illuminate smart streets while saving energy.

Description: Device: Arduino WiFi, detect dawn and twilight with the shield mounted ambient light sensor.

🚩 Build a Water Heating Appliance: Remotely switch heaters and water pumps.

Description: Device: Arduino WiFi, energize shield mounted relays according to a H2M web application.

🚩 Build a <fill in the blank> Appliance: Ideas abound and are found online.

Description: Some popular repositories that may contain useful information include:

<https://developer.mbed.org/cookbook/Student-Projects/> (<https://developer.mbed.org/cookbook/Student-Projects/>)

<https://www.instructables.com/> (<https://www.instructables.com/>)

<https://www.hackster.io/arduino/> (<https://www.hackster.io/arduino/>)

<https://www.hackster.io/NXP/> (<https://www.hackster.io/NXP/>)

<https://www.hackster.io/armmbed/> (<https://www.hackster.io/armmbed/>)

<https://hackaday.io/> (<https://hackaday.io/>)