Lab #4

Introduction to Internet of Things (IoT)

CSCE 5612 - Embedded Hardware/Software Design

Fall 2018

100 Points

Due: 11/15/2018, 11:55 PM

Instructions: TA will be available during the office hours or by an appointment to demonstrate your lab as a team. You are most welcome to demonstrate the lab to the TA as a team before the due date. Please don't wait till the last minute to complete the lab as you may not get the required resources like PCs or boards. Make sure you have a printed lab report (one per team) and turn it in when the demonstration starts. Your lab grader will ask you questions during the demonstration. Questions are worth **20 points** and will be graded according to your answer to the questions. After the demonstration is completed, immediately upload the report and the zipped project to Canvas. **Not following the above instructions could result up to 50% deduction from your lab score**. **Late demonstrations and submissions are not allowed.**

Objectives:

This lab introduces Internet of Things (IoT) and demonstrates an application where an Illuminance sensor (interfaced with ESP8266 NodeMcu) is connected to the Samsung SmartThings hub and the IoT ecosystem is used to control the intensity of a smart LED bulb.

Requirements: (30 Points)

1. Interface the illuminance sensor (TSL2561) with the ESP8266 NodeMcu
2. Convert the ESP8266 NodeMcu as an IoT Thing and connect it to the Samsung SmartThings hub
3. Connect a Sengled Element Color Smart LED bulb to the Samsung SmartThings hub
4. Control the intensity of the Sengled Element color smart LED bulb as follows (a) If the illuminance is less than 30, the bulb should output an intensity of 100% (b) If the illuminance is between 30 and 60, the bulb should output an intensity of 75% (c) If the illuminance is between 60 and 90, the bulb should output an intensity of 25% (d) If the illuminance is greater than 90, the bulb should turn off.

Procedure:

1. A WiFi router is setup in the lab with *ssid:* csce5612, *password:* ntdpf243. The credentials of the router web portal is *username:* admin and *password*: ntdpf243 which can be used to find the IP addresses of the hub and the smart things
2. Download and install the SmartThings mobile application on your smartphone. Create a Samsung account and setup the Samsung Smartthings Hub version 3 following the link https://support.smartthings.com/hc/en-us/articles/205380634-Setting-up-an-account-the-Hub-and-Things
3. Do not connect any Things (devices or sensors) to the hub using the SmartThings app
4. Download and install the SmartThings Classic app on your smartphone. There should be now two apps installed in your smartphone
5. Login into the SmartThings classic app using your initially created Samsung account credentials and you should see the hub setup in the classic app. This lab uses the classic app for the rest of the implementation. The SmartThings app is only used to add the hub into the ecosystem
6. Install the ST_Anything library in Arduino IDE following the instructions in https://github.com/DanielOgorchock/ST_Anything#arduino-ide-setup-instructions
7. Open a new sketch in Arduino IDE for ST_Anything using ESP8266 (File→Sketchbook→Sketches→ST_Anything_I2C_ESP8266WiFi)
8. Modify ESP8266 WiFi information (ssid, password, ip, gateway, subnet, dnsserver, and hubip). This information can be got from the WiFi router
9. Connect the illuminance sensor (TSL2561) to ESP8266 using the I^2C bus
10. Compile the sketch and upload the sketch. Open a serial monitor at 115.2 Kbps and you should see the IP address, MAC, and port details printed on the serial monitor console
11. Login into the Smartthings Groovy Integrated Development Environment (IDE) using the link https://account.smartthings.com/ using the SmartThings app credentials
12. Click on My Hubs to see the details of the hub listed on the website. Click My Device Handlers→Settings and add the Github repositories with Owner as DanielOgorchock and Name as ST_Anything and Branch as master
13. Click on Update from Repo and select ST_Anything (master). Select Parent_ST_Anything_Ethernet and Child Illuminance Sensor and click Execute Update
14. Click on My Devices on SmartThings IDE portal and click on New Device to add the Illuminance sensor (TSL2561)
15. Give a name, device network id (same as name) and select type as Parent_ST_Anything_Ethernet

16. Select the location of the hub and the hub you would like to add the device and finally create the new device
17. Click on the created device and edit the preferences and enter the ip, MAC, and port number of the ESP8266 from the serial monitor and click save to continue. This will also create a child illuminance sensor in the device list
18. Open your SmartThings Classic app on your smart phone and click on My Home→Things and you should see after few minutes the illuminance displayed on the app page
19. Click on refresh (on the app) to get a new updated illuminance value and test to make sure if the illuminance is changing
20. Plug the Sengled Element color smart LED bulb to a power strip using the bulb holder and turn it on and off ten (10) times and the bulb should blink showing that it is ready to be added to the Samsung SmartThings hub
21. On the SmartThings Classic app, click My Home→Things and click Add a Thing and you should see the Sengled LED bulb showing on the list and done to continue. You should now be able to turn on or turn off the LED bulb from the app
22. On the SmartThings IDE website, click on My SmartApps→New SmartApp→From Template and add “Light Up the Night” app and click create
23. The smart app “Light Up the Night” is added to your smart app.
24. Refresh the illuminance sensor and when the illuminance is low (<30) the smart LED light turns on or else should turn off when illuminance is high (>30)
25. Modify the Samsung SmartThings Groovy IDE code to meet the requirement 4
26. SmartThings IDE reference is available at https://docs.smartthings.com/en/latest/
27. Arduino IDE reference is available at https://www.arduino.cc/reference/en/
28. Use the example codes and the above references to meet the requirements and complete the lab
29. Record your observations in the report.

Demonstration: (30 Points)

1. Demonstrate that the illuminance sensor (TSL2561) can be refreshed on the SmartThings Classic app and show that the values of illuminance changes as the ambient light changes
2. Demonstrate that the Sengled Element Color Smart LED bulb can be turned on and off and the intensity can be changed from the SmartThings Classic app
3. Demonstrate that as the illuminance changes the light intensity is changed as per the requirement 4.

Deliverables: (20 Points)

1. Use the lab template posted on the Canvas page

2. Answer all the questions asked in the lab template
3. Type the lab report and turn in the printout of the lab report when the lab is demonstrated as a team
4. After the demonstration is completed, upload the report (one per team) and the zipped project (one per team) to Canvas.