IUPUI ■ CSCI-43300: Internet of Things ■ Spring 2023

Project 1 Assignment

Due Date: Feb. 20, 2023

General Instructions:

1. This assignment is to be carried out by a team of two students or individually.
2. Plagiarism is a violation of the IUPUI academic ethics and will not be tolerated. All material quoted directly from other sources must appear within quotation marks and appropriately referenced; all material that builds upon other sources must be appropriately referenced. This project must be strictly your team's work. Be sure to contact the instructor in case you have any questions as to what constitutes plagiarism or another academic ethics violation.
3. Your report will be graded for content and form. It is important that the report be easy to read and understand (no misspellings, no grammatical mistakes, well labeled figures where appropriate, etc.). The report should contain no hand-written notes or hand-drawn figures.
4. Some milestone(s) might be set by TA to make sure your work in progress.

Objectives:

1. Learn and use a single node system, e.g., Arduino or Raspberry Pi, to build a simple application. (For Raspberry Pi, the projects are Java/Python based. For Arduino, the projects are C based.)

Problem Statement:

Step 1: Install and run the development tools. Run the "Blink" application.

Step 2: Use two LEDs, one resistor, and two buttons to build a simple application. You may have more items in the kit.

- One LED will be lighted on and its brightness will be controlled by a button. Pulse Width Modulation (PWM) can be used to make LED dim. Once the button is pushed, the brightness of the LED should change. At least two levels of brightness are required.
- The other LED will initially blink once one second. It should automatically blink faster/slower after every few seconds. Let us consider to have three different levels of blinking frequencies. When reached the highest/lowest frequency, the LED should go back to the lowest/highest frequency and loop around. Use a button to stop/start the automatic behavior of the LED. That is, once the button is pushed, the LED stops increasing its frequency. When the button is pushed again, the LED resumes to its frequency looping.
- The above two functionalities should share one button (Let's say Button A). The other button B is used to switch the behavior of the button A. For example, suppose A is to increase one LED frequency, when button B is pushed, A should switch to control the other LED's brightness. (This part is not required for an individual project.)

Step 3: Learn to use the temperature sensor. There are different types of temperature sensors. For Arduino, one of the easiest is MCP9700 analog sensor. For Raspberry Pi, the DS18B20 digital temperature sensor will be used. Build an application to turn on the LED if the temperature is higher than a threshold and to turn off the LED if the temperature drops below the threshold.

Report:

Your report should include the following parts: (1) node setup, circuit design and description, (2) application code development, (3) how your applications work for steps 2 and 3, (4) any issues encountered and how they are resolved; lessons learned in your project, and (5) for team project,

indicate which one works on which part of the project. Your report needs to be complete, clear and concise. Submit your report online. Project presentation is required, which includes a live or video demo.

Please include separate folder for circuit diagram/demo video/code/report in a zip file and submit it on Canvas. Zip file naming format: Proj_1_Student1_ Student2.

Grading:

step 2: 40 points

step 3: 30 points

report: 20 points

demo: 10 points

Useful Information and Websites:

a. Raspberry Pi:

- (Get started) https://www.raspberrypi.org/help/videos/#getting-started-with-raspberry-pi
- (Java based library) https://pi4j.com/
- (Press Buttons-Python) https://raspberrypihq.com/use-a-push-button-with-raspberry-pi-gpio/
- (How to use DS18B20, follow step by step) https://www.circuitbasics.com/raspberry-pi-ds18b20-temperature-sensor-tutorial/

  or https://www.hackster.io/weargenius/ds18b20-sensor-interfacing-with-raspberry-pi-using-java-e64893

b. Arduino:

- (Get started) https://www.arduino.cc/en/Guide/HomePage
- (Blink) https://www.arduino.cc/en/Tutorial/Blink
- (Press Buttons) https://www.arduino.cc/en/Tutorial/Button
- (Count button presses) https://www.arduino.cc/en/Tutorial/ButtonStateChange
- (Temperature sensor MCP9700) https://startingelectronics.org/beginners/start-electronics-now/tut15-arduino-serial-thermometer/