CS-350 Emerging Sys Arch & Tech – Project One

Eric Slutz

Southern New Hampshire University

# Project One

To create a working thermostat prototype there are several requirements the microcontroller must meet for it to be considered for use in the prototype. The microcontroller must support I2C for reading from a temperature sensor, GPIO for controlling an LED, the use of GPIO interrupt for two buttons, and UART to simulate sending thermostat data to a server. The microcontroller must also be able to connect to Wi-Fi and have enough RAM and Flash to support the required code.

From its use during this course, it is clear that the TI microcontroller supports all of the peripherals required for the prototype. Additionally, this microcontroller is capable of connecting to a Wi-Fi network, so it would be able to communicate with a server in the cloud over the internet. Lastly, this microcontroller has 256KB of RAM, which is enough to support the needed code. (CC3220S-LAUNCHXL Development Kit | TI.com, n.d.)

The WFI32 series microcontroller development boards from Microchip would also be acceptable for use creating the prototype. These microcontrollers include LEDs and user configurable buttons, an on-board temperature sensor, and is able to connect to Wi-Fi networks and access and communicate with cloud resources. The device supports GPIO, I2C, and UART as well as meeting all of the peripheral and connectivity requirements. This microcontroller also supports enough RAM for the needed code. (*WFI32-IoT Development Board*, n.d.)

The first thing I noticed about Freescale is that it no longer exists; it merged with NXP. I was unable to locate a microcontroller from NXP that meets all of the requirements for the prototype. When searching their site, it also looked like many of the products being offered were from third

parties. There were several cards that seemed to meet most requirements but were missing buttons or LEDs. The one that had those items also had Wi-Fi but was missing the temperature sensor.

All three microcontroller architectures from TI, Microchip, and NXP (Freescale) include software to make it easy to get the device connect to a Wi-Fi network. Once you have made the connection the Wi-Fi controller on the microcontroller uses standard network protocols, such as TCP/IP to communicate over the internet. In this case the intended target is the cloud resources the thermostat prototype needs to report to.

Lastly, the flash and RAM on all three microcontroller architectures support the code in a couple different ways. First the flash serves as a place for the code to be stored so it can be retrieved when it needs to be executed. Second the RAM holds the code that is currently being executed by the microcontroller as well as and variables and their values being used in the code.

# Reference

*CC3220S-LAUNCHXL Development kit | TI.com*. (n.d.). https://www.ti.com/tool/CC3220S-LAUNCHXL

*WFI32-IoT Development Board*. (n.d.). https://www.microchip.com/en-us/development-tool/EV36W50A