CS-350 Final Project

By Thomas Fiske

4/14/2023

The main peripheral used in this project was the TI SimpleLink Wi-fi CC3200S LaunchPad. Connected to that was other peripherals that were used such as the temperature sensor, the buttons on the side, and the LED light. For this thermometer to work, the temperature sensor needs to read the temperature of the room it is currently in, and send that data into the connected device. The same is true of the buttons, they also need to be able to send data to the connected device. The data that they send is if they have been pressed or not. The LED then needs to be told to turn on and off given the data received by the temperature sensor. All these peripherals need to be connected to a device that can take and interpret the data, and then send instructions (LED\_ON) back based on that data. The TI architecture was used in this project. This allowed for a USB connection to a computer that allowed the thermostat to function. TI also supports Code Composer Studio that this project was coding in. Microchip is another integrated developer that has an IDE called MPLAB X, while Freescale/NXP uses Design Studio. Freescale/NXP does support the use of a temperature sensor in its embedded systems (NXP, 2010). Microchip makes their own temperature sensors so they would also support this technology (Microchip, 2022). Both companies also support embedded systems with button functionality. This means that any of these three companies’ architectures and technologies could be used to create a similar thermostat.

The thermometer would need to be connected to the cloud. For this to happen the thermometer needs to connect to the internet which it does through a router. For the thermostat to connect to a router, it will need to support internet functionality, specifically Wi-Fi functionality. The TI board already supports Wi-fi, so that is an architecture that can be used to fulfill that functionality. Freescale/NXP also has systems that use Wi-fi functionality (NXP, 2014), as well as Microchip (2022). This means that any of these architectures could be used to make a thermostat that fulfill the Wi-Fi functionality requirement.

The TI Launchpad has 256KB of RAM and 1MB of executable flash (Texas Instruments, 2019). Both Microchip and Freescale/NXP make products that are similar RAM/Flash. All three companies make products with more Ram/Flash as well.

It seems that every company’s architecture and products can fulfill the needs for this project. Since most of this project has been done already with the TI architecture, I would recommend continuing with this architecture. Microchip does have a very strong programable board as well called the PIC32 WFI32E Curiosity Board. This comes with 32MB of Flash memory (Microchip, 2017) As well as a temperature sensor and Wi-Fi connectivity. This should be a very strong consideration to use if creating a more complex thermometer.

# Sources :

*NXP*. (April, 2010). *Temperature sensor for the HCS08 microcontroller family -* Retrieved April 16, 2023, from https://www.nxp.com/docs/en/application-note/AN3031.pdf

Microchip (2022). *Digital temperature sensors | microchip technology*. Retrieved April 16, 2023, from https://www.microchip.com/en-us/products/sensors-and-motor-drive/temperature-sensors/digital-temperature-sensors

NXP (July, 2014). *Freescale cloud-connected Zigbee Thermostat Getting Started Guide*. Retrieved April 16, 2023, from https://community.nxp.com/pwmxy87654/attachments/pwmxy87654/nxp-designs%40tkb/408/1/Freescale\_Cloud-Connected\_ZigBee\_Thermostat\_Getting\_Started\_Guide%20v0.4.pdf

Texas Instruments. (2019). *LAUNCHXL-F28379D Overview User's guide (rev. C) - ti.com*. Retrieved April 16, 2023, from https://www.ti.com/lit/ug/sprui77c/sprui77c.pdf?ts=1599203131740&ref\_url=https%253A%252F%252Fwww.ti.com%252Ftool%252FLAUNCHXL-F28379D

Microchip. (2017). *PIC32MZ EF Curiosity Development Board user’s guide*. Retrieved April 16, 2023, from https://ww1.microchip.com/downloads/en/DeviceDoc/70005282B.pdf