7-1 Project Submission

Robert Heavner

SNHU

CS-350

Professor Gregori

10/14/2024

Provide the appropriate technical and operational documentation for the task scheduler that you created to satisfy step 7. This should address all the required functionality and be documented according to industry standards describing the algorithm used, its inputs and outputs, and expected results. To excel for this item, you may additionally provide an appropriate state-machine diagram in addition to the other documentation.

- The purpose of the task scheduler is to carry out actions in response to a timer interrupt. With four primary states Idle, Temperature Check, LED Update, and UART Update it functions as a basic state machine. Based on decided timed events, the scheduler transitions between states. When the timer is in the "Idle" state, it waits to start. Following the timer event, it goes to "Temperature Check," where it takes a reading from the sensor to determine the current temperature. After then, it switches to "LED Update," where the heater's LED is either turned on or off based on whether the outside temperature drops below the predetermined level. The last step is "UART Update," where the setpoint, heater status, and current temperature are transferred over UART for simulation. This system receives inputs in the form of timer events and button presses to modify the setpoint; its outputs consist of UART data and LED control.

Explain how the thermostat supports the peripherals used in the project. Make sure that you have included all the required details from the scenario in your report. You should discuss each of the three outlined hardware architectures, including TI, Microchip, and Freescale.

- The thermostat uses a number of add on’s. The temperature sensor is interfaced with via the I2C interface, the buttons and LED (heating control) are managed by the GPIO, and the UART replicates data supplied to the server. The CC3220S, which has the required hardware support for GPIO, I2C, and UART, is used to integrate these peripherals.

Explain how the thermostat connects to the cloud via Wi-Fi. Discuss all three architectures in your work.

- The TI CC3220S, which uses the Wi-Fi functionality, is the method which the thermostat can use a Wi-Fi connection with the cloud. This enables the thermostat to communicate real time information to a distant server for monitoring, such as temperature measurements and heater status. Other modules such as the could be used by the Microchip architecture to manage Wi-Fi connectivity. The KW41Z module from Freescale's Kinetis series provides Wi-Fi capabilities, which can be used to connect to the cloud for comparable features.

Discuss the architecture's Flash and RAM that supports the code. Include all three architectures in your discussion.

- With 256 KB of RAM and 1 MB of Flash, the TI CC3220x platform has more than enough power to handle data storage, run the thermostat code, and control Wi-Fi connection. Larger applications can now have more room thanks to Microchip's microcontrollers with greater memory, such as the PIC32MZ, which has up to 2 MB of Flash and 512 KB of RAM. Up to 1 MB of Flash and 256 KB of RAM are available in the Freescale Kinetis family, which is sufficient to enable the thermostat's features, such as sensor reading, peripheral control, and cloud connectivity.