Jarod Timms

10-13-2022

CS-350

Dr. Kris Jamsa

**Module Seven Project**

The task scheduler operates within a Texas Instruments SimpleLink Development board, specifically a CC3220S model. The code makes use of onboard hardware such as the temperature sensor, side buttons, and the red led. The code initializes drivers for the timers, I2C connections, and UART.

Within the main thread, the timer and value variables are created, and the various hardware components are initialized. The timer itself is initialized with microseconds, so all timer variables are configured on a 100,000 microsecond clock, with the timerPeriod variable set at 100. TimerPeriod acts as a millisecond increment.

Every 200 milliseconds, the model checks the side buttons for flags. Every 500 milliseconds, the temperature is read from the on-board sensor. Every second, the current temperature, setpoint, heat, and runtime duration are printed to console.

The set point is used to preform arithmetic with the active temperature. If the temperature is higher than the set point, the led is triggered to disable. The set point can be increased or decreased by pressing the side buttons.

At the end of each scheduled task, respective timer values reset if their action is triggered. All timers are increased by the timerPeriod variable at the end of the loop.

There are three main vendors we are looking at for microprocessor architecture. These vendors include Texas Instruments, Microchip, and Freescape Semiconductor with a possible extension of NXP. Looking at these three vendors, they all offer an impressive selection of up to tens of thousands of microcontrollers with many offerings support for each of the peripherals we are using in this project. This includes controller architectures which include over 256kb of RAM, GPIO, UART, and I2C connectivity along with integrated timers.

Some potential product choices that we could move forward with would be the Freescale Semiconductors NXP IW612, Microchip PIC32MZ-W1, and the Texas Instruments CC3230S which we are already familiar with. Each of these processors would serve our project well, as all three support embedded WiFi connectivity, though I would suggest we stick with the CC3230S chip as we are already familiar with the architecture.

Citation

*CC3230S and CC3230SF SimpleLink Wi-Fi 2.4GHz Wireless MCU*. CC3230S datasheet | TI.com. (n.d.). Retrieved October 16, 2022, from https://www.ti.com/document-viewer/CC3230S/datasheet

Microchip Technology Inc. (n.d.). *PIC32MZ1025W104132*. Microchip. Retrieved October 16, 2022, from https://www.microchip.com/en-us/product/PIC32MZ1025W104132

*IW612*. IW612: 2.4/5 GHz Dual-Band 1x1 Wi-Fi® 6 (802.11ax) + Bluetooth® 5.2 + 802.15.4 Tri-Radio Solution | NXP Semiconductors. (2022, July 20). Retrieved October 16, 2022, from https://www.nxp.com/products/wireless/wi-fi-plus-bluetooth/2-4-5-ghz-dual-band-1x1-wi-fi-6-802-11ax-plus-bluetooth-5-2-plus-802-15-4-tri-radio-solution:IW612