Final Project: Thermostat Proto-type Report

Brandon Thibeaux

2/20/2022

CS350

Prof. Ricky J. Sethi

South New Hampshire University

The thermostat uses UART and I2C communication. The UART is used to update the server using serial communication, for testing we are watching the updates using PuTTY serial terminal. We use the I2C bus to communicate with the temperature sensor that is built-in to the microcontroller. We also use a timer and timer interrupts to enable our scheduler to keep track of timed tasks. We use GPIO interrupts to allow a user to increment and decrement the set point of when the thermostat turns the heater on or off. For now, we also use the red LED, which is also a GPIO device to represent the heater. So when the heater needs to be on, we just turn the Led on in its place.

Let us discuss three different hardware architectures that could be used for this project. Arduino, ST Electronics Nucleo board, and the TI microcontroller. For this paper, we will specifically look at the Arduino Uno, Nucleo-L467RG, and the TI CC3220S-LAUNCH\_XL boards.

As discussed on the project in the above paragraph, the TI board has a only two UARTs on J6 and J5 columns located on the 20pin connectors. Pins on the columns can be configured to be used through the serial cable or through an external RX and TX pin. The It board only seems to have two I2C ports that are both connected to the built in accelerometer and temperature sensor. We can configure the jumper connectors to allow us access to external I2C devices using these two buses. Additionally we also seem to have SPI connectivity, built in SimpleLink Wi-Fi device to allows developers to enable their projects to connect to WiFi. Then the data being transported to the connect Wi-Fi can be accessed through phones are computer devices. This device also uses an ARM Cortex Processor.

The STM32 Nucleo-L476RG has a lot of pins. It has a lot of bult in peripherals, but this board does not have Wi-Fi. It does however have a pin layout exactly like the Arduino Uno to make Arduino shields easier to be used by the microcontroller and it has two other extension connectors. The Nucleo-board has many IO pins and any of them can be interrupt pins. This allows for very complex systems to be developed that is interrupt driven. See the resource section of this paper for more details because the board is highly customizable that it all cannot fit on this paper. But this board does have several UART channels, a few SPI channels, a few I2C channels, several channels for ADC pins, 8 timers, 2 watchdog timers, and an ARM Cortex processor.

Lastly, the Arduino has 14 IO pins which 6 can be pwm pins, 6 analog pins, uses the ATmega328P processor, only can use two pins as interrupt pins, has 1 I2C channel and 1 SPI channel, and one serial/UART RX/TX channel. As you can see this one does lack a lot of hardware options as compared to the others. But What makes everyone love this board is due to its simplicity and get started quick design. Even the IDE, community, and libraries made for this board are very user friendly and vast. This does have a trade off that you are heavily limited to what you are able to do with this option though.

To get more in detail lets talk about the three architecture’s RAM and FLASH storage. The TI board has 1 MB of flash memory and 256 KB of RAM. The Nucleo-L476RG has 512KB of Flash memory and 96 KB of SRAM. And the Arduino has 32KB of flash memory and 2KB of SRAM memory. Flash memory is non-volatile memory that holds your program represented as a binary set of instructions. The RAM is volatile memory that stores data related variables, pointers, and arrays to your program, RAM is deleted and erased when the power goes out. As you can see the TI clearly has the better specs as it holds bigger programs and allows that program to have more moving parts and attributes to it.

In conclusion for the comparison, the TI board has the advantage for this project simply because it has built in Wi-Fi components. Also if we needed room to store a very big and complex program this microcontroller could adapt to that. Though a close second place would be the Nucleo board because this board would allow far more peripherals and pins to be used which one of them could be a Wi-Fi or ethernet module. The advantage that Arduino gives is that the community and resources available will quickly get your project built. But as you build your project them ore complicated and complex it gets the harder you will find that it is to maintain the project using this board. This board’s advantage lies within the prewritten library’s available for it.

Let’s go back to our project. How can we enable this prototype to be integrated with a cloud service via Wi-Fi? Using the TI board and the current program that was built for the thermostat prototype. We could capture the data as normal, but instead of sending the data string over UART we send it over Wi-Fi using the SimpleLink Wi-Fi component built into the CC3220S board. Over Wi-Fi we could send that data to a cloud server for the data to be parsed and processed and accessed by other remote devices like a phone, smart TV, or PC devices. The cloud would essentially act as a mediator between the sensor and the clients.

Of we were to use the other two boards we would have to buy Wi-Fi/ Ethernet modules. Once connected to Wi-Fi we would follow the same method of sending message packets trough a socket using Wi-Fi that contain the data transmitted from our program. A Good module to possibly use for these two microcontrollers is the ESP8266 which connects to the microcontroller using serial communication and forwards data from the microcontroller to the Wi-Fi-router to send data to the cloud.

Resources:

Amazon. (n.d.). *Amazon.com: Hiletgo 3pcs ESP8266 nodemcu CP2102 ESP-12E ...* HiLetgo 3pcs ESP8266 ESP-01 Serial WiFi Wireless Transceiver Wireless Module Development Board LWIP AP+STA Compatible with Arduino. Retrieved February 19, 2022, from https://www.amazon.com/HiLetgo-Internet-Development-Wireless-Micropython/dp/B081CSJV2V

*Arduino® Uno R3*. (n.d.). Retrieved February 19, 2022, from https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf

askSensors, & Instructables. (2020, July 21). *Connecting Arduino WIFI to the cloud using ESP8266*. Instructables. Retrieved February 19, 2022, from https://www.instructables.com/Connecting-Arduino-WiFi-to-the-Cloud-Using-ESP8266/#:~:text=Connecting%20Arduino%20WiFi%20to%20Cloud%20Using%20ESP8266%201,the%20Code.%205%20Run%20the%20Code.%20See%20More.

*Meet the CC3220x LaunchPad Development Kit (CC3220S-LAUNCHXL and CC3220SF-LAUNCHXL)*. Ti devtools. (n.d.). Retrieved February 19, 2022, from https://dev.ti.com/tirex/explore/node?node=ANkGhhEkpfMuOBJ20Qo2Jw\_\_fc2e6sr\_\_LATEST

*STM32 nucleo F401RE Development Board*. Microcontrollers Lab. (2020, May 28). Retrieved February 19, 2022, from https://microcontrollerslab.com/stm32-nucleo-board-pinout-features-applications-programming-peripherals/

Texas Instruments . (2017, February). *CC3220 SimpleLink™ Wi-Fi® and Internet of Things Technical Reference Manual*. Texas Instruments. Retrieved February 19, 2022, from https://www.tij.co.jp/jp/lit/ug/swru465/swru465.pdf?ts=1645306197690&ref\_url=https%253A%252F%252Fwww.bing.com%252F