ECNS-424

IDL #3

9 April 2021

Brandon Empie

I2C Synchronous Serial Communication

Objective: Integrated circuit communication creates a method to combine services/devices into one embedded system with multiple functions. The purpose of this lab is to using I2C take advantage of 23Kx8 Serial EEPROM, TC-74 temperature sensor, and Blink-M smart LED on the BulldogPIC++. The state of switch S2 is to set the color of the Blink-M smart I2C LED with intensity based on POT's position. An interrupt occurring every 100mS is to display the current color being shown as well as a readout of the current value of the POT. The current, high, and low temperature is to be read with the TC-74 in the interrupt and displayed to the LCD by cycling switch S3. The high and low temperatures are to be stored in EEPROM and switch S4 is required to clear the temperatures in EEPROM to the current temperature by the user.

Methods: To accomplish the many tasks the main loop of the program checks the switches state and makes necessary updates based on user input and sends all current data to the LCD. The interrupt was designed to "collect" all data required every 100mS. The baud rate was set at 50kHz using equation baud rate = 8MHz/(4*(39+1)). I2C was handled using library functions to consolidate code required and configured to only be used inside the interrupt "Blink". The high and low recorded temperature was stored at address location 0x0330 and 0x0320 in EEPROM to be evaluated against the current temperature in the interrupt.

Results: As the POT increased to max (255) the Blink-M intensity increased to max and was cycled between red, green, and blue using switch S2. At 24 degrees Celsius the current temperature was read in and saved during program initialization to the EEPROM. As the temperature increased to 27 degrees and decreased to 23 degrees the LCD updated its display as switch S3 was cycled. By pressing switch S4 all three settings returned to room temperature at 24 degrees as expected. This process was verified by single stepping the program in MPLAB X IDE environment, verifying variable values in memory, using math calculation with a calculator, and power cycling the BulldogPIC++.

Conclusion: The program required multiple moving pieces and had to be laid out based on each task required. The programmer quickly realized that when the master is using I2C to communicate to its slave devices it is critical that it not be interrupted and have enough time to complete each sequence to avoid complete program failure.