



# TEMPERATURE SENSOR AND VISUALIZER WITH PIC24 MICROCONTROLLER

TRUMAN BROWN

# OVERVIEW

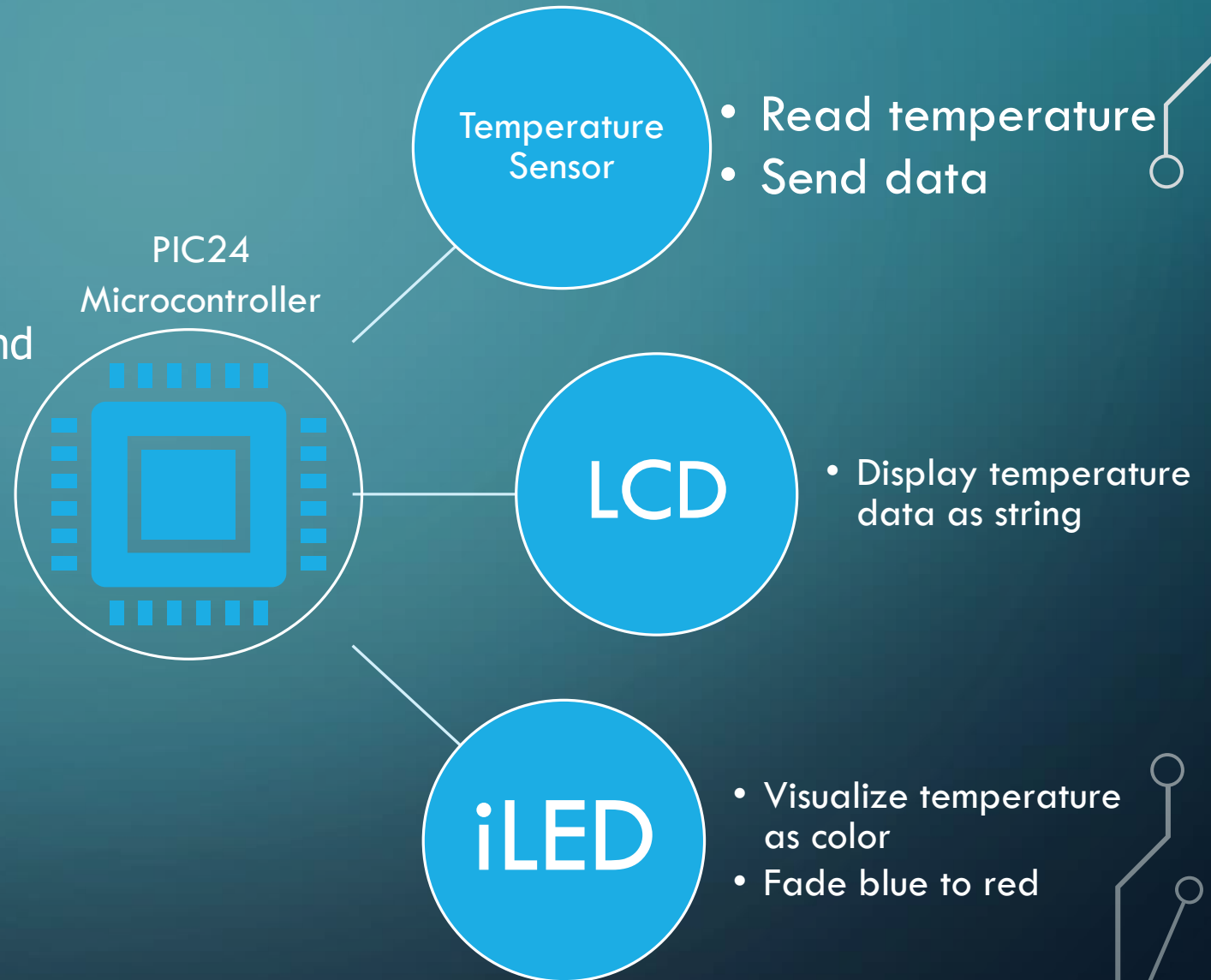
- Goal:

- Assemble temperature sensor, LCD, and iLED to work together

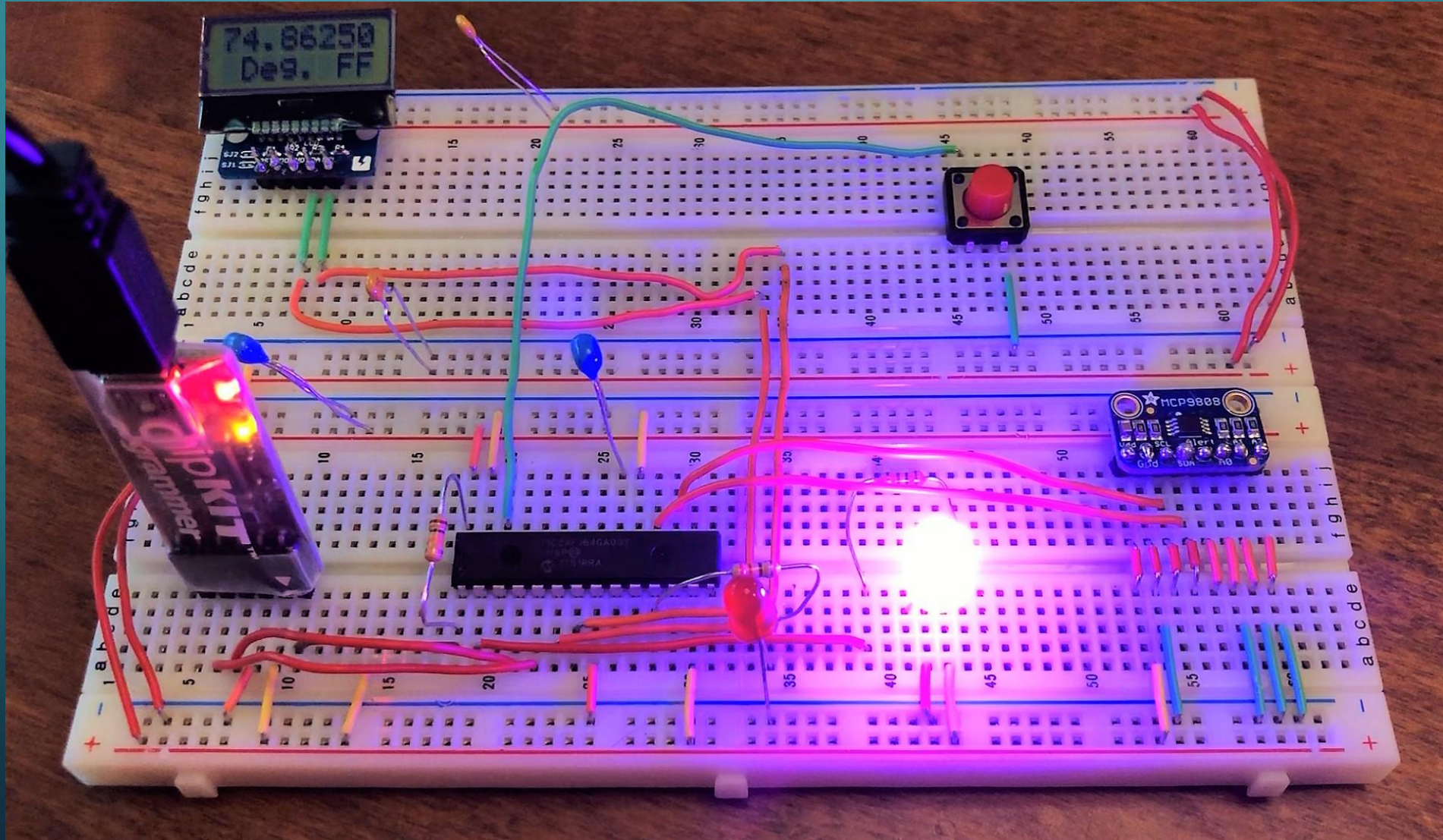
- MCP9808 High Accuracy I2C Temperature Sensor Breakout Board
- Akizuki/Sitronix Controlled LCD
- WS2812 iLED

- Helpful additions:

- Push-button switch
- Debugging LED



# CIRCUIT



# LIBRARIES NEEDED

## Assembly Library

- Various delays (100 $\mu$ S, 1ms)
- Write Pulses functions on RAO
  - 0.35 $\mu$ s in 1.25 $\mu$ s period
  - 0.70 $\mu$ s in 1.25 $\mu$ s period

## iLED Library

- Setup the LED
  - I/O, 16MHz, etc.
- Variable delay
  - # of ms delay
- Write a color to the LED
  - Controlled with red, green blue values
  - 0-255 value for each color



# LIBRARIES NEEDED (CONT.)

## LCD Library

- Setup the LCD
  - I2C, IF cleared, etc.
  - Minimum frequency of 100 kHz
- Write commands to the LCD
- Initialize the LCD with series of commands
- Set the cursor on the LCD
  - Row, column
- Write a single character to the LCD
- Write a string/constant array of characters to LCD

## Temperature Sensor Library

- Initialize the temperature sensor
  - Minimum frequency of 100 kHz
  - Initialized to idle condition
- Initiate the start condition for the data transfer protocol
- Initiate the stop condition for the data transfer protocol
- Write data to the temperature sensor
  - Return either acknowledge status
- Read digital temperature data from sensor
- Configure the sensor with series of writes
- Initialize the double click detector on the button-switch

# HOW ALL THE LIBRARIES WORK TOGETHER IN MAIN

- Call all setup functions from all libraries (setupLED(), setupLCD(), lcd\_init(), etc.)
- Initialize necessary variables
  - Char array to store floating point value of temperature
  - Two arrays of size 255 to store “checkpoints” of temperatures to compare to sensor reading
    - {70.0, 69.85, 69.70, 69.55,.....,(70-i\*0.15)}, {70.0, 70.15, 70.30, 70.45,.....,(70+i\*0.15)}
- While(1) loop
  - Read temperature sensor data into two bytes as required by MCP9808 sensor using temperature sensor library functions
  - Display temperature value on LCD using LCD library functions
    - Switch between Celsius and Fahrenheit
  - Fade iLED between blue and red in correspondence to temperature readings from sensor
    - Uses iLED library functions to write the color to the LED.

# PROBLEMS ENCOUNTERED AND THEIR SOLUTIONS

Getting the temperature sensor configured correctly

- The datasheet along with the LCD I2C experience helped

Converting the ambient temperature sensor register into a decimal value

- Studying two's complement again to better understand how to convert

Fading the iLED smoothly between red and blue

- Switched while loops of checkpoints to an array of checkpoints

