Project Final Report

Precipitation Predictor

21 April 2011
Heidi Benson

Project Abstract

The final product displays humidity (%), temperature (C), and barometric pressure (Pa) on the LED array using a humidity sensor and barometer/thermometer. In addition, the temperature is shown with an arrow controlled by a magnetic step motor. One of the three measurements is displayed by pressing one of the three buttons on the ZNEO board.

I created libraries for each of the devices. The HIH-4030 humidity sensor communicates measurements with analog voltage; the BMP085 barometric pressure sensor communicates using I^2C; and the step motor, SMCC-547, was controlled using GPIO pins. Both sensors required additional math to convert the values into actual readings. The humidity sensor required only a simple slope equation, $y= mx+b$. The BMP085 required reading calibration coefficients before the first measurement and using these in long calculations.

Status

The project did not proceed as planned. Originally, the project was going to track pressure and humidity, and with them, predict precipitation. I was going to watch for sudden changes of pressure within a couple hours and use the humidity to gauge whether rain was likely. Because of consistent trouble communicating with the BMP085 using I^2C until very shortly before the due date, I was unable to fully implement all my plans. I had only two days so I had to cut rain prediction entirely, and chose to leave it as a very simple weather station.

When I demoed, I successfully showcased the functionality of all external hardware.

Specification

<u>**Hardware:**</u>
Platform: based on the Zilog Z16 series board with the ZNEO microprocessor. (professor)
Capabilities: GPIO, I^2C, interrupts, ADC, Timers, LED, Buttons, UART (debugging)
External Hardware:

- Honeywell HIH-4030 (www.sparkfun.com)
- Bosch BMP085 (www.sparkfun.com)
- SMCC-547 (professor)
- 4.7 k resistors (professor)

Software Used/Written:

- Drivers for the HIH-4030, BMP085, and SMCC-547
- String display for the LED array
- Button library to control functionality using interrupts
  - Button 1 read the ADC and displayed humidity
  - Button 2 read and displayed present temperature, also moving the step motor arrow for additional visual indication of temperature.
  - Button 3 read and displayed present barometric pressure.

For debugging purposes, I enabled the UART serial port.

Implementation & Construction

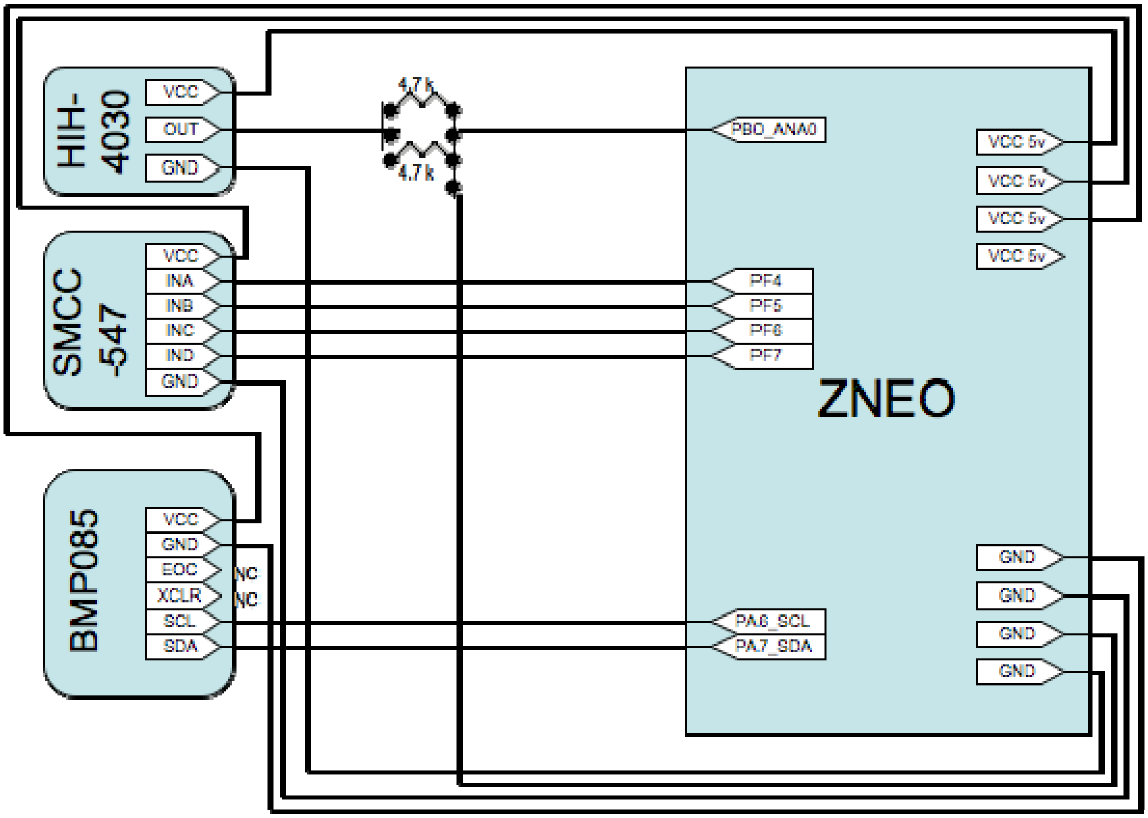

Figure 1: Hardware diagram

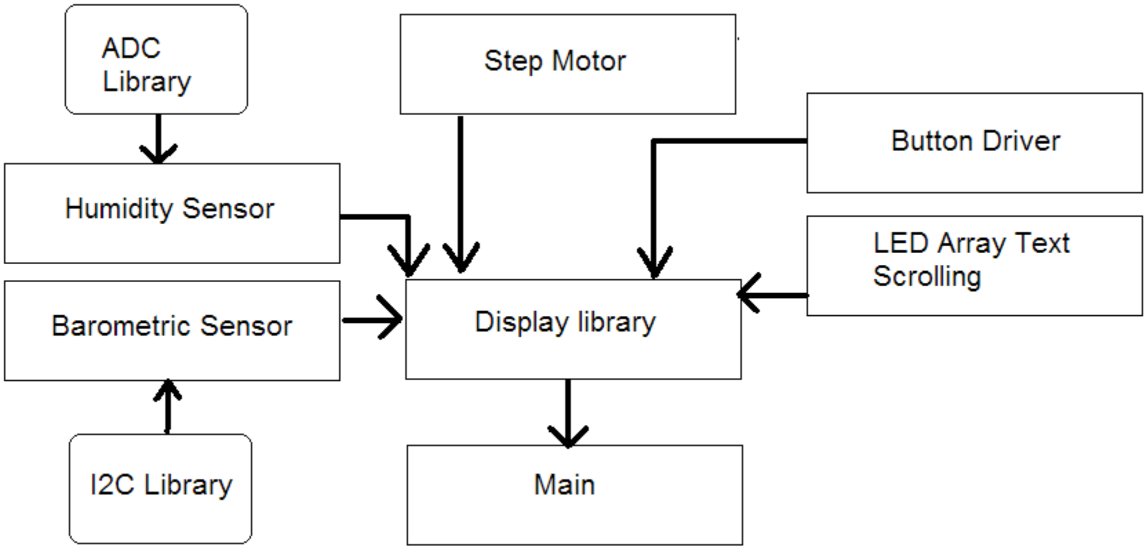

Figure 2: Software Diagram

Milestones*:

1. Wrote ADC library
2. Wrote SMCC-547 (step motor) driver
3. Wrote HIH-4030 (humidity) driver
4. Wrote frame for Display library
5. Wrote BMP085 (barometric pressure) library
6. Finished Display library
7. Some wires were soldered
8. Demo

* Button driver and LED scrolling were already written from the labs.

Test and Verification:

During my development for the external hardware, I often employed the oscilloscope. I also used the voltmeter extensively for the HIH-4030.

Once things were connected, I used the UART serial to printf() messages to Putty and performed the tried and true method: Push the Buttons Over and Over. I changed my code as appropriate, when the issue was with the code. When my results were spotty, and I was holding the wires to try and keep them consistent, the professor suggested soldering the wires because they were evidently loose. All the readings came in correctly after that. I verified the readings against weather websites on the Internet.

QuickTime™ and a
decompressor
are needed to see this picture.

Figure 3: Photograph of final project.

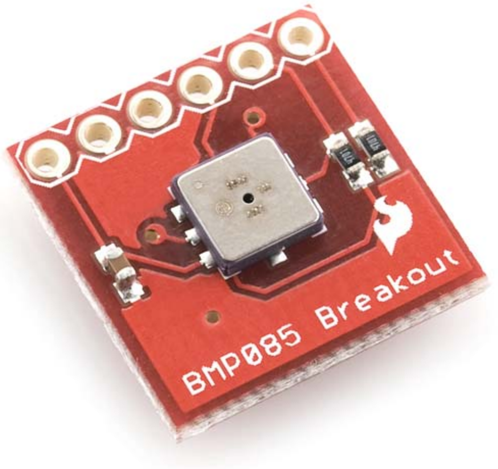

Figure 4: Bosch BMP085 on breakout board. (Picture used without permission. http:\\www.sparkfun.com\)

Figure 5: Honeywell HIH-4030 on breakout board. (Picture used without permission. http:\\www.sparkfun.com\)

Retrospective

My most important design decision by the end was how to showcase each device in a complete manner that makes sense since I hadn't enough time to implement rain prediction. Using the buttons to initiate and display present measurements was the simplest method.

From this project I learned that I should look at soldering sooner rather than later, since this would have saved me a week of loose wires. I've also learned that unless or until I gain a greater understanding of datasheets, I should pick external devices more on how understandable their datasheets are when possible.

If I had known previously what I know now, I would have searched for devices with different communication methods. ADC is very simple and so far my favorite method of communication, with the UART as a close second. I^2C seems difficult for me, for whatever reason, and I would have preferred to avoid it if possible. Overall though, I don't think I would have changed the project concept at all, and since I hadn't yet learned about I^2C, I had no way of knowing that I would have so much difficulty with it.

Attachments

Source code
BMP085 Datasheet
SMCC-547 Datasheet
HIH-4030 Datasheet