8

Temperature Reading Using 1-Wire Bus

8.1 Introduction

In this experiment, you are going to write a program which will display the temperature on the LCD screen in every 3 seconds as shown in the Fig 8.1. In order to do this, you will use the digital thermometer DS18B20 on the experiment board, which is connected to the microcontroller as shown in the Fig. 8.2. Students are required to read the datasheet of DS18B20 in order to get ready to the experiment.

Fig. 8.1. A Sample Temperature Measurement Display

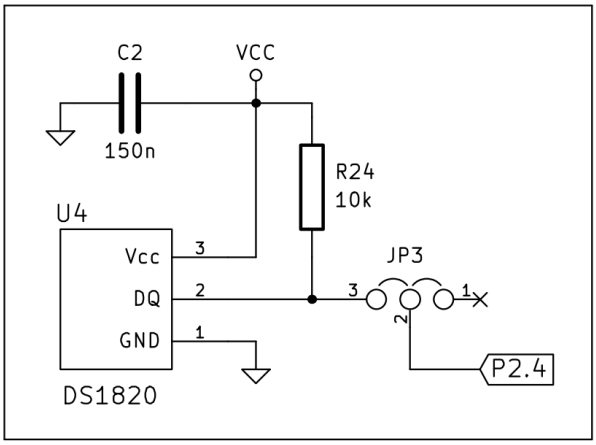

Fig. 8.2. Connection between DS18B20 and MSP430

8.2 Background Information

The DS18B20 thermometer is provides digital temperature data in Celcius degree unit in varying resolutions (i.e., 9-bit to 12-bit resolutions). These resolutions allow application developers to measure the temperature with 0.5 ° C, 0.25 ° C, 0.125 ° C and 0.0625 ° C resolutions. For example, the temperature measurement shown in Fig 8.1 shows that the temperature is 24.8125 ° C where the LSB 0 refers to 2^{-4}, bit 1 refers to 2^{-3}, bit 2 refers to 2^{-2}, and so on. Note that selection of the resolution affects the timings since the thermometer will require longer time intervals to convert the temperature data to the specified resolution. As you will see in the datasheet, higher the resolution, longer the time it takes for DS18B20. You should remember this when you will command DS18B20 to convert the temperature data. After your command, you have make sure that your program waits for the corresponding amount of time when DS18B20 is doing the conversion.

The DS18B20 thermometer communicates with the external world via a single wire. In order to configure the thermometer and read temperature, one has to send commands or read the temperature data bit-by-bit using only a single wire. This behavior is achieved via 1-Wire Communication Protocol. In this protocol, the single wire is called as the 1-Wire Bus System where DS18B20 acts as a slave and the microcontroller acts as the bus master.

Accessing DS18B20 is achieved via 3 steps:

- **Step 1:** Initialization
- **Step 2:** ROM command
- **Step 3:** Function command

Note that these steps are required to be followed one by one *for each access* to DS18B20. Please check the datasheet for the contents of these steps, ROM and function commands.

Up until this experiment, you have always used general purpose input output pins either as an input or as an output. However, in this experiment the 1-wire communication protocol will ask you to use a single pin of a single port as input for some time and output for some other time. Another very crucial thing is that up until this experiment you were not paying much attention to the timings. However in this experiment you must pay attention to the timings and delays which can affect the performance of your program severely.

8.3 Experiment

In this experiment, you will configure the thermometer and read the temperature data in every 3 seconds and display the data in the LCD or 7-segment display. In order to implement the experiment, you may write following subroutines:

- **InitT:** The initialization procedure where you will implement the reset and presence pulse operations.
- **WriteCMD:** Subroutine which will be used to write either ROM commands (e.g., Skip ROM) or function commands (e.g., ConverT, Read Scratchpad) to the 1-wire bus bit-by-bit. Since commands will have 1s and 0s, you will have to pay attention to how to write 1 and 0 to the bus.
- **ReadBit:** Subroutine to read a bit from the 1-wire bus. Reading from the bus is a little-bit more complex than writing to the bus since the time interval to read the correct data from the bus is limited, which means if

you do not read on time, then you can read 1s all the time. You will use this subroutine to read the temperature data.

Just to remind you that, you should read the datasheet before coming to the experiment sessions. Otherwise, you may not be successful during the experiment.

8.4 Report

Prepare your report by using the guidelines and the report template which are posted on Ninova e-Learning System.

During the experiment, please do not forget to take notes about the critical points of the implementations in order to write a proper report for the experiment. Additionally, if there were any complications which affect your performance during the experiment, please also indicate these difficulties in your report.