The University of Jordan
Embedded Systems Lab
Fall 2020/2021

Names	ID
عبد الرحمن خالد حامد	0164872
محمد القواسمه	0145838
ابراهيم الصمادي	0138565
سليمان الشطناوي	0155582

Temperature Control System in Datacentre

1: Introduction

In this project, you will be using a PIC16F877A microcontroller to design a simple control system to be used in a server room in some datacentre. Basically, the room is equipped with two huge 220-Volt DC fans and it is required to control the temperature inside the room in a power- efficient approach through adjusting the speed of the fans according to the room temperature.

2: Methodology

2.1. Configuration

The approach for this project was using the port-b change interrupt for the modes of the pic and the HOLD mode as well; the pic was configured using the ADCON0 to make the pic operates on ($F_{osc}/8$) and making the LSB of port-a input (digital input), port-d configured to output and connected directly to the LCD as well as port-c.

The Pulse Width Modulation (PWM) module was configured using TM2 by putting the pre-scale of it to (16) and setting the bits of CCP1CON into PWM mode and capturing it.

2.2. Auto mode

This mode was set to be the default mode, once the pic is started the program will operates on this mode, in this mode the pic will first start reading the value that comes from pin RA0 (the result of KTY81), it will read it as an analog value then by calling the next sub-routine (READ) it converts it to digital in a range from (0 to 1023) as for this state we come to approach to split this range into four quarters, table [1] shows those them.

Range	Quarter
0-255	First
255-512	Second
512-778	Third
778-1023	Fourth

Table [1]: The four quarters of the digital values

The next sub-routine which will calculate the temperature and convert that range into a new range from (0 to 40), to achieve this purpose two registers will be used (ADRESL and ADRESH), ADRESL is an 8-bit register that will be used to store the value of the temperature, but the other register (ADRESH) will be used to point at what quarter we are in ; For example, let the value before calling (CAL_TEMP) sub-routine is equal (511) after calling it the value will be stored in a temporary register and divided on (25) since we need to know which quarter we are in at that value, after finding the result of the division it will be stored in another temp-register and we call the MULTIPLY sub-routine that will find what quarter we are in and add that value to the temp-register that have the range from (0 to 10) and store the result in the TEMPARATURE register.

2.3. Manual mode

This mode is the same as the auto mode but instead of just reading from RA0 for the temperature, it will also read from RA1 for the speed of the Dc-motors; But unlike the auto mode this mode will only operates when pushing the required button and change port-b value which will cause an interrupt (port-b change interrupt).

3: Results

As a result, for all of this work the pic operates smoothly with a decent number of delays to display the requirements on the LCD and run the fans as it should be; the figures below will show the proteus file on the running state in the auto and manual mode as well as the fans.

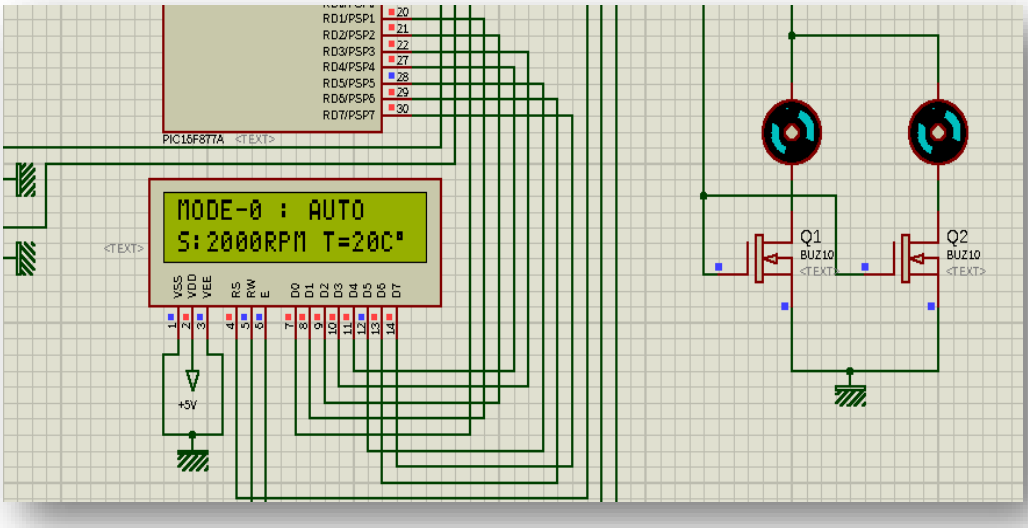


Figure [1]: proteus file running in the AUTO mode

From Fig. [1] we can see that the LCD show what mode we are at and also it displays the data into 8-bits into 2-lines with 5x7 Dots , also the S=2000 RPM which indicates that the ADRESH=2 (in the THIRD quarter) and after displaying that on the LCD it will transfer that value into the motors.

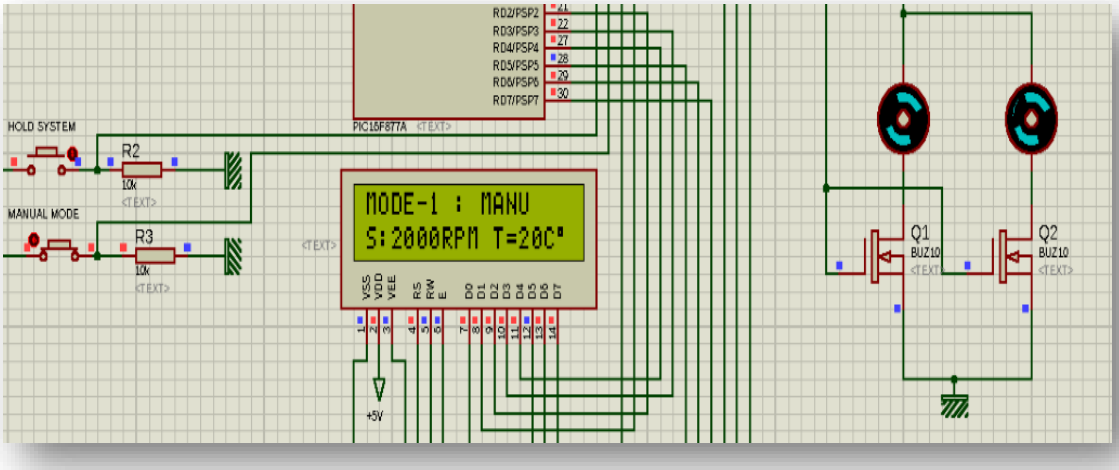


Figure [2]: proteus file running in the MANUAL mode

References

[1]: (LibStock - KTY-81 Temperature Sensor (Binary Search Algorithm), 2020) Libstock.mikroe.com. 2020. *Libstock - KTY-81 Temperature Sensor (Binary Search Algorithm)*. [online] Available at: <https://libstock.mikroe.com/projects/view/1400/kty-81-temperature-sensor-binary-search-algorithm> [Accessed 26 December 2020].

[2]: Electronics Hub. 2020. *Digital Temperature Sensor Circuit Using 8051 & AVR Microcontrollers*. [online] Available at: <https://www.electronicshub.org/digital-temperature-sensor-circuit/> [Accessed 26 December 2020].

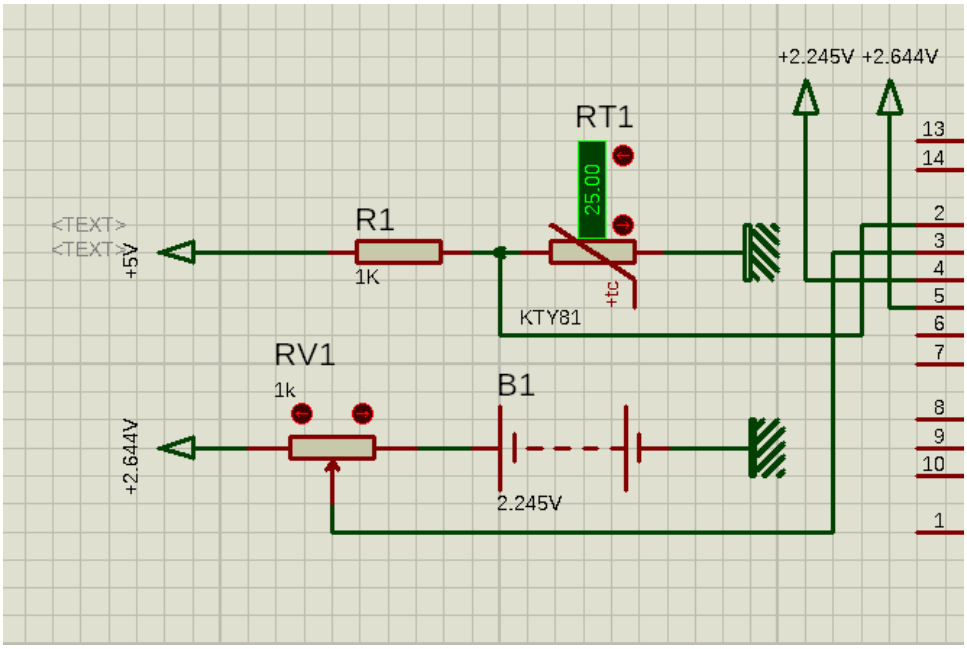
[3]: Circuit Digest. 2020. *Digital Thermometer Using LM35 And 8051 Microcontroller*. [online] Available at: <https://circuitdigest.com/microcontroller-projects/digital-thermometer-using-lm35-8051> [Accessed 26 December 2020].

[4]: 2020. [online] Available at: <https://embedded-ju.ucoz.com/> [Accessed 26 December 2020].

[5]: Circuit Digest. 2020. *Generating PWM Using PIC Microcontroller With MPLAB And XC8*. [online] Available at: <https://circuitdigest.com/microcontroller-projects/pic-microcontroller-pic16f877a-pwm-tutorial> [Accessed 26 December 2020].

Appendices

- KTY81 AND CONTROL KNOB:



- PUSH BUTTONS: