**T.C.**

**BAHÇEŞEHİR UNIVERSITY**

**FACULTY OF ENGINEERING AND NATURAL SCIENCES**

**DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING**

**PROJECT REPORT**

**Humidity Controller**

**EEE3205 –Microcontrollers**

**Course Project**

**Furkan Tekden**

**-1904016**

**Ömer Ayar**

**-2101069**

**İSTANBUL, Jan2022**

# TABLE OF CONTENTS

TABLE OF CONTENTS (right-click and select "Update Field") ii

1 OVERVIEW 1

1.1 Problem Statement and Objectives 1

1.2 Background Information 1

2 METHODOLOGY 2

2.1 Project Design 2

2.2 Project components 2

2.3 Final product and results 2

2.4 Code 2

3 CONCLUSION 3

4 REFERENCES 4

# OVERVIEW

## Problem Statement and Objectives

Humidity controllers monitor and maintain proper humidity levels in environmental test applications, food storage areas, and electronic equipment rooms, Humidity controllers monitor and maintain proper humidity levels in environmental test applications, food storage areas, and electronic equipment rooms.

## Background Information

Increasing the relative humidity can make people feel warmer in the winter, so you don't need to boost the heating. That also means the HVAC system doesn't have to work as hard to achieve temperature set points, so you're saving on wear and tear

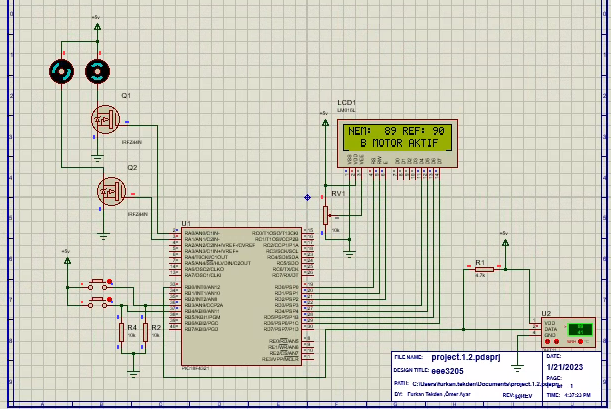


Figure 1. Circuit schematic in Proteus

# METHODOLOGY

## Project Design

The hummidty controller fixes the humidity to the desired level. When the reference point humidity level is reached, the Controller turns off the motor. The other motor will then automatically turn on and off as required by the Controller to maintain the desired humidity level.

## Project components

DHT11;The DHT11 is **a basic, ultra low-cost digital temperature and humidity sensor**. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and spits out a digital signal on the data pin (no analog input pins needed).

IRFZ44 ;it is one of the N Channel Mosfet Types and the TO220 sheath is tandoor. IRFZ44 is of type N Channel Power Mosfet, they do switching and power balancing work in the project.

LCD1 LM016;Liquid crystal display is a display technology based on the principle that the electrically polarized liquid transmits light in a single phase and can be seen with a polarization filter added in front of it. Liquid crystal displays have replaced the formerly used vacuum fluorescent displays with their low energy consumption.shows humidity and reference values in the project

DC MOTOR ;The machine that converts direct electrical current into mechanical energy is called DC motor. The necessary energy conversion is provided by the windings and permanent magnets in the DC motor.one motor reduces the measured excess humidity while the other engine increases the measured under humidity in project.

THE PIC18F4321; features a C compiler-friendly development environment, 256 bytes of EEPROM, self-programming, an ICD, 2 capture/compare/PWM functions, 13 channels of 10-bit Analog-to-Digital (A/D) converter, 2 comparators, the synchronous serial port can be configured as either 3-wire Serial Peripheral Interface (SPI™) or the 2-wire Inter-Integrated Circuit (I²C™) bus and Enhanced Universal Asynchronous Receiver Transmitter (EUSART), 8 MHz internal RC oscillator and advanced low power oscillator controls.It performs the operations of the code we have written and runs and controls the other elements of the circuit as a whole.

POT-HG ;Potentiometers are resistors whose value can be changed by external physical interventions. The more powerful potentiometers and those used in higher current circuits are called rheostats.

RESİSTOR;Resistors are used to limit the current in electrical circuits to keep it at a certain value.

## Final product and results

Above a certain humidity, our humidity control device starts the motor that reduces the humidity. if it is below a certain humidity, it starts the motor that increases the humidity and these work according to the references we have written to the microcontroller.

## Code

Share your code here. Try to explain your codes with comments using //.Code Exam

|  |
| --- |
| #include <nem.h>  #define LCD\_RS\_PIN PIN\_D0  #define LCD\_RW\_PIN PIN\_D1  #define LCD\_ENABLE\_PIN PIN\_D2  #define LCD\_DATA4 PIN\_D3  #define LCD\_DATA5 PIN\_D4  #define LCD\_DATA6 PIN\_D5  #define LCD\_DATA7 PIN\_D6  #fuses NOMCLR INTRC\_IO  #use delay(clock = 8000000)  #include <lcd.c>  #use fast\_io(B)  #BIT Data\_Pin = 0xF81.0  #BIT Data\_Pin\_Direction = 0xF93.0  #define DEBOUNCE\_DELAY 200  unsigned int16 reference\_humidity = 90;  int increase\_button = pin\_b3;  int decrease\_button = pin\_b4;  unsigned int16 humidity, temperature;  int fan\_one = pin\_a0;  int fan\_two = pin\_a1;  short Time\_out;  unsigned int8 T\_byte1, T\_byte2, RH\_byte1, RH\_byte2, CheckSum;  int debounce(int pin) {  if (input(pin)) {  delay\_ms(DEBOUNCE\_DELAY);  if (input(pin)) {  return 1;  }  }  return 0;  }  void start\_signal() {  Data\_Pin\_Direction = 0;  Data\_Pin = 0;  delay\_ms(25);  Data\_Pin = 1;  delay\_us(30);  Data\_Pin\_Direction = 1;  }  short check\_response() {  delay\_us(40);  if (!Data\_Pin) {  delay\_us(80);  if (Data\_Pin) {  delay\_us(50);  return 1;  }  }  return 0;  }  unsigned int8 Read\_Data() {  unsigned int8 i, k, \_data = 0;  if (Time\_out) break;  for (i = 0; i < 8; i++) {  k = 0;  while (!Data\_Pin) {  k++;  if (k > 100) { Time\_out = 1; break; }  delay\_us(1);  }  delay\_us(30);  if (!Data\_Pin) bit\_clear(\_data, (7 - i));  else {  bit\_set(\_data, (7 - i));  while (Data\_Pin) {  k++;  if (k > 100) { Time\_out = 1; break; }  delay\_us(1);  }  }  }  return \_data;  }  void main() {  setup\_oscillator(OSC\_8MHZ);  setup\_adc\_ports(NO\_ANALOGS);  lcd\_init();  lcd\_putc('\f');    while (TRUE) {  delay\_ms(1000);  Time\_out = 0;  Start\_signal();    if (check\_response()) {  RH\_byte1 = Read\_Data();  RH\_byte2 = Read\_Data();  T\_byte1 = Read\_Data();  T\_byte2 = Read\_Data();  Checksum = Read\_Data();  if (CheckSum == ((RH\_Byte1 + RH\_Byte2 + T\_Byte1 + T\_Byte2) & 0xFF)) {  temperature = T\_Byte1;  humidity = RH\_Byte1;    lcd\_putc('\f');  lcd\_gotoxy(1, 1);  lcd\_putc("HUMIDITY: ");  printf(lcd\_putc, "%3Lu", humidity);  lcd\_putc("% REF:");  printf(lcd\_putc, "%3Lu", reference\_humidity);  if (debounce(increase\_button)) {  reference\_humidity++;  if (reference\_humidity > 100) reference\_humidity = 100;  }  if (debounce(decrease\_button)) {  reference\_humidity--;  if (reference\_humidity < 0) reference\_humidity = 0;  }  lcd\_gotoxy(1, 2);  if (humidity > reference\_humidity) {  output\_high(fan\_one);  output\_low(fan\_two);  lcd\_putc("FAN A ON");  } else {  output\_low(fan\_one);  output\_high(fan\_two);  lcd\_putc("FAN B ON");  }  lcd\_gotoxy(1, 3);  lcd\_putc("TEMP: ");  printf(lcd\_putc, "%3Lu", temperature);  lcd\_putc("C");  }  }  }  } |

# CONCLUSION

While carrying out the project, we learned that humidity is a very important factor for people to work under normal conditions in our world and where technologies are developed.At the same time, we have strengthened the working principles of the microcontroller in technical terms,Some of the stages we can develop in our project can store previously measured data, measure the average relative humidity over a period of time, and display the relative humidity information.

# REFERENCES

1. <https://www.microchip.com/en-us/product/PIC18F4321>
2. <https://en.wikipedia.org/wiki/Humidistat>
3. <https://stackoverflow.com/questions/46648968/interfacing-sensirion-sht31-with-pic18f2420>
4. <https://microcontrollerslab.com/digital-humidity-sensor-using-pic-microcontroller/>