INTELLIGENT HUMIDISTAT

MICROPROCESSORS AND INTERFACING

CS/ECE/EEE/INSTR-F241

Submitted by: Group 69

Group Members

Tekumal Saisaketh	-	2019A8PS0636G
Sai Mahesh Nimishakavi	-	2019A7PS0153G
A.R Jithin	-	2019A8PS0581G
Neelareddy siddhartha reddy	-	2019AAPS0242G
A.G.S.M Bhaskar	-	2019A7PS0124G
S.G.S Surya	-	2019AAPS0321G

Date: 19/4/2021

ACKNOWLEDGEMENT

The successful completion of this report required a lot of guidance and support that we received throughout this project. We would like to extend our deepest gratitude to the Instructor-in-Charge (IC) of this course, Prof. K.R. Anupama for giving us this opportunity to work on such an interesting assignment.

We would like to extend our gratitude to all the lab instructors who have helped us to get command over assembly language programs.

CONTENTS

PROBLEM STATEMENT

A humidistat is supposed to be reset based on the outside temperature--as the outside temperature drops, so should the humidity level inside the house. The goal of this project is to create a humidistat that detects outside temperature and adjusts humidity accordingly. Two sensors are required: outside temperature and inside humidity. Output is provided via a simple relay with the humidifier (presumably on the furnace) being on or off. Also, readings from the humidity and temperature sensors must be displayed on an LCD display. The entire system can be turned on or off using a single switch.

USER REQUIREMENTS & TECHNICAL SPECIFICATIONS

Design a system that senses the temperature and humidity and turns on the humidifier when temperature is less than or more than our requirement

The Technical specifications are as follows:

- Temperature range is –40 to 60 Degree Celsius
- Relative Humidity range 0 to 100 %RH

ASSUMPTIONS

1) Room is small so only one temperature and one humidity sensor are required.

2) Resolution of 1°C and 1% is required in temperature and humidity sensors respectively.

3) There is a linear relationship between temperature and humidity.

4) Temperature has to be sensed every 2 minutes and the display should be updated

SYSTEM DESCRIPTION

The humidistat is designed to adjust the humidity level in a room based on the outside temperature. The humidifier can be turned on if the actual humidity is less than the optimal humidity for the current outside temperature. The humidifier's role is to increase the humidity in the room. If the outside temperature rises after the optimum humidity has been reached, the humidifier is enabled.

Two sensors are used for this: an outside temperature sensor and an inside humidity sensor.

The temperature sensors are mounted outside the room and are exposed to the elements. The humidity sensors are installed inside the bed. The humidity sensor tests the humidity of the humidistat in percent relative humidity. Analog output is provided by the sensors. A/D converters are used to transform these outputs to digital form.

The ADC is configured so that the output varies from 00h to 64h, with each 1°C rise in temperature or 1 percent increase in RH resulting in a 01h increase in output.

LIST OF COMPONENTS USED

Chip No	Quantity	Chip	Purpose
8086	1	Microprocessor	C.P.U
2716	2	ROM	Where code resides
6116	2	RAM	Used for stack and temporary data
8255	2	Programmable Peripheral Interface	Provides I/O ports for the other devices
8253	1	Programmable Interval Timer	To generate clock input for ADC
8259	1	Programmable interrupt Controller	Management of hardware interrupts
LM016L	1	16*2 alphanumeric LCD	Displays the current temperature and humidity
ADC0808	1	Analog to digital converter	Converts the analog volage to its digital equivalent
74LS245	2	8-bit-Bidirectional Buffer	Buffering data bus
74LS373	3	8-bit octal latches	Latching the address bus

OTHER HARDWARE USED

Logic Gates – These are mainly used to build encoding logic for memory and I/O interfacing.

Solid-state Relay – It acts as a key to turn on high-voltage equipment.

LED – It is used to indicate whether the humidifier is turned on or off.

Potentiometers – These are used to simulate sensor data.

Switches – These are used to switch the device and the LCD on and off.

Address Map

Memory Map

ROM1 – 00000_H – $00FFF_H$

RAM 1– 01000_H – $01FFF_H$

ROM2 – $FF000_H$ – $FFFFF_H$

I/O Map

8255_1 – 00_H – 06_H

8255_2 – 08_H – $0E_H$

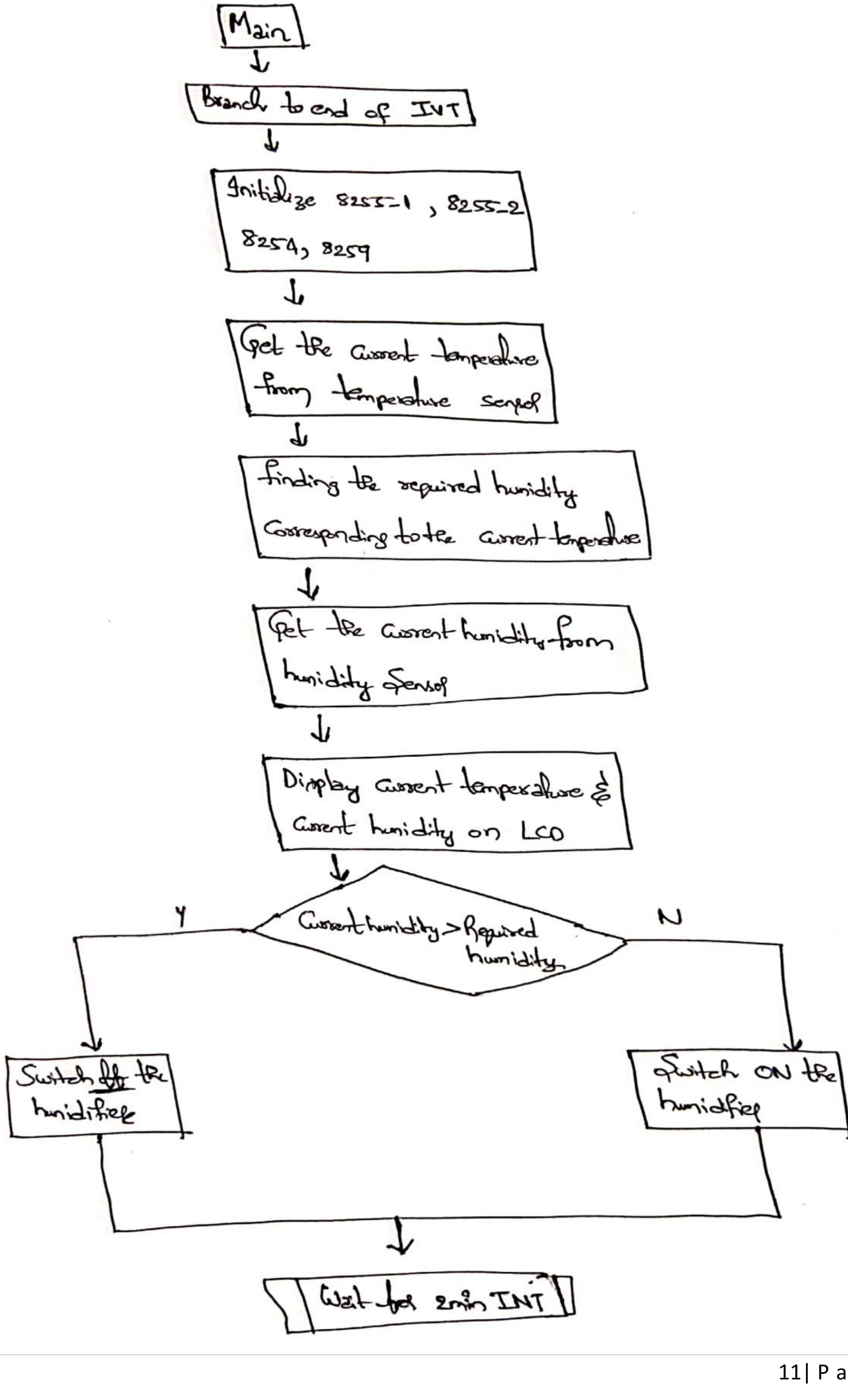
8254 – 10_H – 16_H

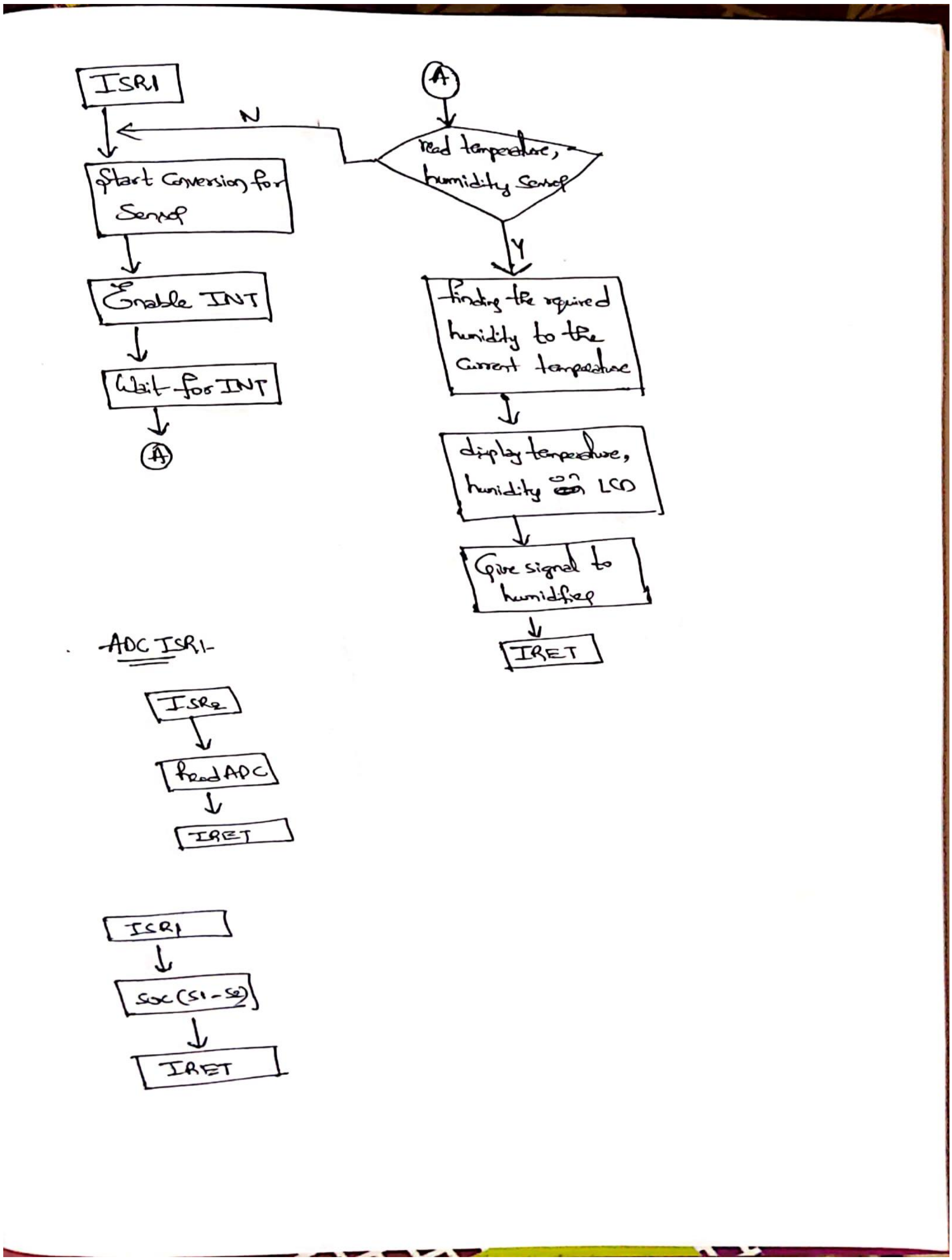
8259 – 18_H – $1A_H$

PORT ADDRESS MAPS

CHIP	A7	A6	A5	A4	A3	A2	A1	A0
8255_1								
00h	0	0	0	0	0	0	0	0
06h	0	0	0	0	0	1	1	0
8255_2								
08h	0	0	0	0	1	0	0	0
0Eh	0	0	0	0	1	1	1	0
8254								
10h	0	0	0	1	0	0	0	0
16h	0	0	0	1	0	1	10	
8259								
18h	0	0	0	1	1	0	0	0
1Ah	0	0	0	1	1	0	1	0

FLOW CHART


Main
Branch to end of IVT
Initialize 8255-1, 8255-2
8254, 8259
Get the current temperature from temperature sensor
finding the required humidity corresponding to the current temperature
Get the current humidity from humidity sensor
Display current temperature & current humidity on LCD
Current humidity > Required humidity
Y
N
Switch off the humidifier
Switch ON the humidifier
Wait for 2min INT


ISR1
N
Start Conversion for Sensor
Enable INT
Wait for INT
A
A
read temperature, humidity Sensor
Y
finding the required humidity to the current temperature
display temperature, humidity on LCD
Give signal to humidifier
IRET
ADC ISR1-
ISR2
Read ADC
IRET
ISR1
SOC (S1-S2)
IRET

Proteus Implementation Variations of Justification

1. Using a 2-minute delay as 8259 does not function in proteus – EOC is used as NMI and the Timer Int is replaced with a 2-minute programmed delay.

2. ROM in just 00000 – this is because proteus updates the reset address.

3. Used 8253 instead of 8254 since 8254 is not present in Proteus.

4. The clock is set to 2 MHz since the clock created for the 8086 needs a long rise and fall time of the clock.

The ADC clock would be just 500KHz, which is fine since the maximum clock that can be given is 1 MHz

5. 2732 is used in place of 2716, which is not present in Proteus.

6. Using a gate-based memory circuit – does the same function as LS 138 here.

7. Temperature and Humidity Sensors are replaced by potentiometers – as all sensors are missing in Proteus 8. 8259 is missing – the same reason as in point 1.

List of Attachments

1) Complete Hardware Real World Design – Drawing.pdf

2) Manuals

    A) MCP9701A – Temperature Sensor

    B) ADC0808

    C) HTG3500 - Humidity Sensor

3) Proteus file – project.dsn

4) EMU8086 ASM file. – code.asm

5) Code for on paper design – code2.asm

6) Binary file after assembly –code.bin