

ระบบควบคุมสภาพแวดล้อมในห้องปิด

Chamber Control



โดย

นาย ชีรยุทธ ชนาศรีมงคลกุล

นาย อนุมาศ มั่งมี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เลขหมู่.....

เลขทะเบียน 50191

วัน เดือน ปี 27 เม.ย. 2547

b.....

i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ภายนอก

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2545

ภาควิชา วิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมสภาพแวดล้อมในห้องปิด

ผู้จัดทำ

1.ธีรยุทธ ธนาศรีมงคลกุล 42010151

2.อนุมาศ มั่งมี 42010429



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมสภาพแวดล้อมในห้องปิด

ธีรยุทธ ธนาศรีมงคลกุล

อนุมาศ มั่งมี

รศ.ดร.วันชัย ธีรจุฑา อาจารย์ที่ปรึกษา

ปีการศึกษา 2545

บทคัดย่อ

โครงการระบบควบคุมสภาพแวดล้อมในห้องปิดนี้ มีจุดประสงค์ในการควบคุมอุณหภูมิและความชื้นให้ได้ตามต้องการซึ่งจะใช้ Microcontroller ตระกูล PIC เบอร์ 16F877 เป็นตัวควบคุมการทำงานและมีแผงปุ่มกด (Keypad) รับการป้อนค่า Input แสดงค่าออกที่จอ LCD การควบคุมอุณหภูมิ นั้น จะอยู่ในช่วงตั้งแต่อุณหภูมิห้อง ถึงอุณหภูมิ 70 องศาเซลเซียส ซึ่งการเพิ่มอุณหภูมินั้น จะใช้ตัวกำเนิดความร้อนที่อยู่ในเตาอบ เป็นตัวเพิ่มอุณหภูมิ และลดอุณหภูมิโดยใช้พัดลมขนาดเล็กดูดอากาศที่ร้อนภายในเตาอบออกเพื่อให้อุณหภูมิในเตาอบลดลง และจะใช้ตัว IC DS1820 เป็นตัววัดอุณหภูมิ ส่วนในการควบคุมความชื้น จะใช้ตัวทำความชื้นชนิด Ultrasonics จากนั้นใช้พัดลมดูดความชื้นเข้าไปในระบบเพื่อสร้างความชื้น และใช้ตัว Humidity (SHT1x) เป็นตัววัดความชื้นสัมพัทธ์ ภายในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CHAMBER CONTROL

Threerayut Thanasrsmongkonkul

Anumas Mungmee

Associate Professor Vanchai Riewruja Advisor

2002

Abstract

In this project, the chamber control is presented. This project has an objective to Control temperature and humidity. PIC 16 F877 is the controller. The extranal keypad is used to receive the input value and the LCD is used to display the output value. The temperature in the room can be varied from the room temperature to 70 degree celcius. The heater in oven is used to increase temperature and the exhaust fan is used to decrease temperature. DS 1820 is use to measure temperature. Humidity are maked Ultrasonics Humidity and are measured by SHT 1x.

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของโครงงาน	1
1.2 ขอบเขตการศึกษา	1
1.3 วัตถุประสงค์ของโครงงาน	2
1.4 ขั้นตอนการศึกษา	2
บทที่ 2 ทฤษฎีและหลักการ	4
2.1 Microcontroller (PIC 16F877)	4
2.1.1 ความเร็วของ PIC	4
2.1.2 หน่วยความจำของ PIC	4
2.1.3 สถาปัตยกรรมของ PIC	5
2.1.4 ชนิดของ PIC	5
2.2 Heater	7
2.3 ไอซีตรวจจับอุณหภูมิ DS1820	7
2.4 ตัวทำความชื้นชนิดอัลตราโซนิก (ULTRASONIC HUMIDIFIERS)	8
2.5 หน้าจอแสดงผล (LCD DISPLAY)	9
2.5.1 รายละเอียดเกี่ยวกับโมดูล LCD	9
2.5.2 โมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด (LCD16x1)	11
2.5.3 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD	12
2.5.4 จังหวะการทำงานของ LCD โมดูล	13
2.6 คีย์แพด หรือ สวิตช์เมทริกซ์ 4x3จุด	13
บทที่ 3 การออกแบบและการสร้าง	15
3.1 หลักการทำงาน	15
3.2 ส่วนประกอบของโครงงาน	15
3.2.1 ห้องปิดจำลอง	18
3.2.2 ตัวทำความร้อน (heater)	19
3.2.3 ตัวทำความชื้นชนิดอัลตราโซนิก(Ultrasonic Humidity)	19
3.2.4 ตัววัดอุณหภูมิ(DS1820) และ ตัววัดความชื้นสัมพัทธ์(SHT1x)	20
3.2.5 พัดลมระบายอากาศ(exhaust fan) และ พัดลมดูดอากาศ(inhaust fan)	21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6 ส่วนควบคุมและประมวลผล (Control Panel)	21
3.3 วงจรควบคุมการทำงานของระบบ	22
3.3.1 วงจรรับค่าและแสดงผล	22
3.3.2 วงจรควบคุมตัวทำความร้อนและพัดลมระบายอากาศ	24
3.3.3 วงจรแสดงการควบคุมตัวทำความร้อน (heater)	25
3.4 การควบคุมอุณหภูมิและความชื้น	26
3.4.1 การควบคุมอุณหภูมิ	26
3.4.2 การควบคุมความชื้น	26
3.5 โปรแกรมควบคุมการทำงาน	27
3.5.1 โปรแกรมหลัก	27
3.5.2 โปรแกรมย่อยควบคุมอุณหภูมิ	29
3.5.3 โปรแกรมย่อยควบคุมความชื้น	30
3.5.4 โปรแกรมย่อยควบคุมอุณหภูมิและความชื้น	32
3.6 การใช้งานโปรแกรมควบคุม	34
3.7 ลำดับขั้นตอนการใช้งาน	35
บทที่ 4 การทดลองและผลการทดลอง	36
4.1 การทดลองที่ 1 การหาประสิทธิภาพของตัวทำความร้อนที่อยู่ในเตาอบ(heater)	36
4.1.1 ขั้นตอนการทดลอง	36
4.1.2 สรุปผลการทดลอง	37
4.2 การทดลองที่ 2 การควบคุมอุณหภูมิ ที่ 70 องศาเซลเซียสโดยป้อนพิกัดแรงดันที่แตกต่างกัน	38
4.2.1. ขั้นตอนการทดลอง	38
4.2.2 สรุปผลการทดลอง	38
4.3 การทดลองที่ 3 การควบคุมการเพิ่มอุณหภูมิที่อุณหภูมิเป้าหมายต่างๆ	39
4.3.1 ขั้นตอนการทดลอง	39
4.3.2 สรุปผลการทดลอง	39
4.4 การทดลองที่ 4 การควบคุมการลดอุณหภูมิที่อุณหภูมิเป้าหมายต่างๆ	40
4.4.1 ขั้นตอนการทดลอง	41
4.4.2 สรุปผลการทดลอง	42
4.5 การทดลองที่ 5 การหาประสิทธิภาพของตัวทำความร้อน ชนิด อัลตราโซนิก	42
4.5.1 ขั้นตอนการทดลอง	42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 สรุปผลการทดลอง	42
4.6 การทดลองที่ 6 การควบคุมความชื้นสัมพัทธ์ที่ความชื้นเป้าหมายต่างๆ	43
4.6.1 ขั้นตอนการทดลอง	43
4.6.2 สรุปผลการทดลอง	46
4.7 การทดลองที่ 7 การควบคุมอุณหภูมิและความชื้นสัมพัทธ์ ให้ได้ผลตามต้องการ	46
4.7.1.ขั้นตอนการทดลอง	47
4.7.2 สรุปผลการทดลอง	52
บทที่ 5 บทสรุปและวิจารณ์	53
5.1 ปัญหาที่เกิดขึ้น	53
5.2 แนวทางการพัฒนา	53
5.3 สรุปและวิจารณ์	53
กิตติกรรมประกาศ	55
เอกสารอ้างอิง	56
ภาคผนวก ก	
ภาคผนวก ข	



สารบัญรูป

รูป	หน้า
รูปที่ 2.1 microcontroller ตระกูล PIC	4
รูปที่ 2.2 OTP (One Time Programmable)	6
รูปที่ 2.3 EPROM (Erasable Programmable ROM)	6
รูปที่ 2.4 EEPROM /Flash (Electrically Erasable Programmable ROM)	7
รูปที่ 2.5 เต้าเสียบไฟฟ้า	7
รูปที่ 2.6 การจัดขาของ DS1820	8
รูปที่ 2.7 ตัวทำความชื้นชนิดอัลตราโซนิก (ultrasonic humidifiers)	9
รูปที่ 2.8 โค้ดแอสเซมบลีการทำงานของโมดูล LCD แบบอักษร	10
รูปที่ 2.9 รูปร่างและการจัดขาโมดูล LCD แบบอักษร	11
รูปที่ 2.10 วงจรของสวิตช์แบบต่อเข้ากับไฟเลี้ยงและกราวด์	13
รูปที่ 2.11 วงจรสวิตช์แบบเมตริกซ์หรือคีย์แพด	13
รูปที่ 3.1 บล็อกไดอะแกรมของหลักการทำงาน	16
รูปที่ 3.2 แสดงด้านหน้าของโครงงาน	17
รูปที่ 3.3 แสดงด้านข้างของโครงงาน	17
รูปที่ 3.4 แสดงด้านบนของโครงงาน	18
รูปที่ 3.5 แสดงด้านหน้าของห้องปิดจาลอง	19
รูปที่ 3.6 แสดงตัวทำความร้อน(heater)	19
รูปที่ 3.7 แสดงตัวทำความชื้นชนิดอัลตราโซนิก(Ultrasonic Humidity)	20
รูปที่ 3.8 แสดงตัววัดอุณหภูมิ(DS1820) และ ตัววัดความชื้นสัมพัทธ์(SHT1x)	20
รูปที่ 3.9 แสดงพัดลมระบายอากาศ(exhaust fan) และ พัดลมดูดอากาศ(inhaust fan)	21
รูปที่ 3.10 แสดงส่วนควบคุมและประมวลผล (Control Panal)	22
รูปที่ 3.11แสดงวงจรรับค่าและแสดงผล	23
รูปที่ 3.12วงจรควบคุมตัวทำความชื้นและพัดลมระบายอากาศ	24
รูปที่ 3.13วงจรแสดงการควบคุมตัวทำความร้อน (heater)	25
รูปที่ 3.14 แสดงการควบคุมความชื้น	26
รูปที่ 3.15 แผนผังแสดงการทำงานหลัก	28
รูปที่ 3.16 แผนผังแสดงการควบคุมอุณหภูมิ	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.17 แผนผังแสดงการควบคุมความชื้น	31
รูปที่ 3.18 แผนผังแสดงการควบคุมอุณหภูมิและความชื้นสัมพัทธ์	33
รูปที่ 3.19 แสดงส่วนติดต่อกับผู้ใช้งาน	34
รูปที่ 4.1.1 แสดงวงจรตัวหรีไฟ	36
รูปที่ 4.1.2 แสดงผลการทดลองหาประสิทธิภาพของตัวทำความร้อน (heater)	37
รูปที่ 4.2 แสดงผลการควบคุมอุณหภูมิ ที่ 70 องศาเซลเซียส โดยป้อนพิกัดแรงดันที่ แตกต่างกัน	38
รูปที่ 4.3 แสดงผลการควบคุมอุณหภูมิที่อุณหภูมิเป้าหมายต่างๆ	40
รูปที่ 4.4 แสดงผลการทดลองการควบคุมการลดอุณหภูมิที่อุณหภูมิเป้าหมายต่างๆ	41
รูปที่ 4.5 แสดงผลการหาประสิทธิภาพของตัวทำความชื้นชนิด อัลตราโซนิก	42
รูปที่ 4.6 แสดงผลการทดลองการควบคุมความชื้นที่ความชื้นเป้าหมาย 50 % RH (การทดลองครั้งที่ 1)	44
รูปที่ 4.7 แสดงผลการทดลองการควบคุมความชื้นที่ความชื้นเป้าหมาย 50 % RH (การทดลองครั้งที่ 2)	45
รูปที่ 4.8 แสดงผลการทดลองการควบคุมความชื้นที่ความชื้นเป้าหมาย 70 % RH (การทดลองครั้งที่ 1)	45
รูปที่ 4.9 แสดงผลการทดลองการควบคุมความชื้นที่ความชื้นเป้าหมาย 70 % RH (การทดลองครั้งที่ 2)	46
รูปที่ 4.10 แสดงผลการทดลองการควบคุมอุณหภูมิและความชื้นสัมพัทธ์ที่มีค่าเท่ากัน กรณีป้อนค่าอุณหภูมิเป้าหมายที่ 50 องศาเซลเซียสและความชื้นสัมพัทธ์ เป้าหมายที่ 50 % RH	48
รูปที่ 4.11 แสดงผลการทดลองการควบคุมอุณหภูมิและความชื้นสัมพัทธ์ที่มีค่าเท่ากัน กรณีป้อนค่าอุณหภูมิเป้าหมายที่ 65 องศาเซลเซียสและความชื้นสัมพัทธ์ เป้าหมายที่ 65 % RH	49
รูปที่ 4.12 แสดงผลการทดลองการควบคุมอุณหภูมิที่มีค่ามากกว่าความชื้นสัมพัทธ์ โดยป้อนค่าอุณหภูมิเป้าหมายที่ 60 องศาเซลเซียสและความชื้นสัมพัทธ์ เป้าหมายที่ 15 % RH	50
รูปที่ 4.13 แสดงผลการทดลองการควบคุมอุณหภูมิที่มีค่าน้อยกว่าความชื้นสัมพัทธ์ โดยป้อนค่าอุณหภูมิเป้าหมายที่ 35 องศาเซลเซียสและความชื้นสัมพัทธ์ เป้าหมายที่ 50 % RH	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงการ

ในระบบอุตสาหกรรมสิ่งแวดล้อมภายนอกเช่น อุณหภูมิ ความดัน ความชื้น เป็นปัจจัยสำคัญที่มักทำให้เกิดปัญหาเกี่ยวกับเครื่องมือและเครื่องจักรต่างๆเสมอ โดยเฉพาะอย่างยิ่งวงจรอิเล็กทรอนิกส์ที่จะต้องทำงานในสภาพแวดล้อมของโรงงานอุตสาหกรรมที่มีทั้งอุณหภูมิ ความชื้นและแสงสว่างซึ่งมีผลกระทบโดยตรงดังนั้นจะต้องมีการ ทดสอบวงจรอิเล็กทรอนิกส์ว่าสามารถทำงานได้ในช่วงอุณหภูมิเท่าไร และจะเริ่มเกิดค่าผิดพลาดที่อุณหภูมิเท่าไร ซึ่งความชื้นก็ต้องมีการ ทดสอบเช่นกัน เพื่อให้ทราบว่าต้องควบคุมสภาพแวดล้อมภายในระบบเป็นเท่าไรจึงทำให้วงจรอิเล็กทรอนิกส์สามารถทำงานได้ อย่างสมบูรณ์แบบรวมไปถึงการควบคุมสภาพแวดล้อมภายในขอบเขต การผลิตก็มีความสำคัญเป็นอย่างมากเช่นกัน เนื่องมาจากผลิตภัณฑ์หลายชนิดในขั้นตอนการผลิตต้องมีการควบคุมอุณหภูมิ ความชื้น เช่น เครื่องอุปโภค บริโภคหลายๆชนิดโดยเฉพาะอย่างยิ่ง อาหารสด หรือ อาหารสำเร็จรูปต่างๆ ซึ่งในรูปแบบของโครงการนี้ ได้มีการจำลองสภาพแวดล้อมที่มีการควบคุมอุณหภูมิและความชื้นให้ได้ตามที่ต้องการ

1.2 ขอบเขตการศึกษา

1. ศึกษาการควบคุมอุณหภูมิและความชื้นภายในห้องปิดโดยใช้ Microcontroller ตระกูล PIC เบอร์ 16F877
2. ศึกษาวงจรการทำงานในการเพิ่มอุณหภูมิ ลดอุณหภูมิ เพิ่มความชื้น และลดความชื้น รวมทั้งศึกษาการทำงานของตัวตรวจจับอุณหภูมิ(DS1820) ตัวตรวจจับความชื้น (SHT1x)และการทำงานของพัดลมระบายอากาศ
3. ศึกษาการเขียนโปรแกรมภาษา C เพื่อใช้ในการควบคุม Microcontroller (PIC 16F877)

1.3 วัตถุประสงค์ของโครงการ

การศึกษาระบบควบคุมสภาพแวดล้อมในห้องปิด เป็นการศึกษาถึงแบบจำลองที่จะต้องควบคุม อุณหภูมิความชื้นเป็นปัจจัยที่สำคัญต่อระบบการผลิตเป็นอย่างมากต้องมีการควบคุมให้อยู่ในช่วงที่กำหนดเพื่อจะได้ผลิตภัณฑ์ที่มีคุณภาพตามคุณลักษณะที่ต้องการ ซึ่งจากขอบเขตของการศึกษาโครงการแบบจำลองนี้สามารถเพิ่มอุณหภูมิและลดอุณหภูมิได้ตามต้องการ เมื่อเทียบกับอุณหภูมิห้อง ซึ่งเป็นข้อได้เปรียบเมื่อเทียบกับการควบคุมอุณหภูมิใน zone การผลิต หากต้องมีการเปลี่ยนแปลงอุณหภูมิเพื่อเปลี่ยนแปลงผลิตภัณฑ์ที่จะผลิตจะต้องมีการปรับตั้งค่า zero, span ของตัว sensor และต้องสั่งให้ตัวทำอุณหภูมิทำอุณหภูมิให้ได้ตามต้องการซึ่งยุ่งยากและเสียเวลาเป็นอย่างมาก หรือแม้แต่การควบคุมความชื้นซึ่งสามารถเพิ่มและลดได้ตามต้องการ เป็นประโยชน์อย่างมากต่ออุตสาหกรรมที่ต้องมีการควบคุมความชื้นที่แม่นยำเช่นอุตสาหกรรมส่งออกข้าวเปลือก, ถั่วลิสง, ถั่วเหลือง หากมีการควบคุมอุณหภูมิและความชื้นที่ดีก็จะได้ผลิตภัณฑ์ที่มีคุณภาพและตรงตามความต้องการของลูกค้า

1.4 ขั้นตอนการศึกษา

1. ค้นหาข้อมูลเกี่ยวกับโครงการเช่นข้อมูลเกี่ยวกับการควบคุมสภาพแวดล้อมทั้งในด้านการควบคุมสภาพแวดล้อมทั้งในด้านการควบคุมอุณหภูมิ และ ความชื้น ที่อยู่ในระบบปิดรวมทั้งปัจจัยและผลกระทบต่างๆที่มีผลกระทบต่อสภาพแวดล้อมในระบบปิด ค้นหาข้อมูลเกี่ยวกับวัสดุอุปกรณ์ที่จะนำมาใช้จัดทำ โครงการเช่นนี้เช่น ตัวไมโครคอนโทรลเลอร์ (PIC 16F877) ตัววัดสัญญาณที่ใช้ในการวัดอุณหภูมิ (DS 1820) และ sensor ที่ใช้ในการควบคุมความชื้น (Humidity) รวมทั้งตัวหน้าจอแสดงผล (LCD) และแผงปุ่มกด (Keypad)
2. จัดหาวัสดุ, อุปกรณ์ในการทำ โครงการ และศึกษาทำความเข้าใจวัสดุและ อุปกรณ์ พร้อมทั้งทดสอบการทำงานเพื่อพร้อมที่จะนำมาใช้งาน เช่นการทดสอบ ตัวหน้าจอแสดงผลตามที่ต้องการ และมีการลองป้อนค่า input ที่ Keypad และแสดงผลที่หน้าจอ LCD เป็นต้น
3. วางแผนงานและออกแบบวงจร hardware ของโครงการชุดนี้ทั้งในส่วนของ hardware การควบคุมอุณหภูมิและ hardware ของการควบคุมความชื้น
4. ประกอบ hardware ในส่วนของการควบคุมอุณหภูมิตามวงจร และดำเนินการทดสอบให้ทำงานตามที่ได้มี การวางแผนไว้
5. ศึกษาการเขียนโปรแกรมโดยใช้ภาษา C เพื่อไปควบคุม microcontroller (PIC 16F877)

6.เขียนโปรแกรม software ของการควบคุมอุณหภูมิให้เป็นไปตาม flowchart ที่ได้มีการวางแผนเอาไว้

7.ทดสอบการทำงานของโครงงานในส่วนของการควบคุมอุณหภูมิทั้ง hardware และ software เพื่อให้ทำงาน ได้อย่างสมบูรณ์

8. ประกอบ hardware ในส่วนของการควบคุมความชื้นตามวงจรที่ได้มีการออกแบบไว้ และดำเนินการทดสอบให้ทำงานตามที่ได้มีการวางแผนไว้

9. เขียนโปรแกรม software ให้เป็นไปตาม flowchart ที่ได้มีการวางแผนเอาไว้

10.ทดสอบการทำงานของโครงงานในส่วนของการควบคุมความชื้นทั้ง hardware และ software เพื่อให้ทำงาน ได้อย่างสมบูรณ์

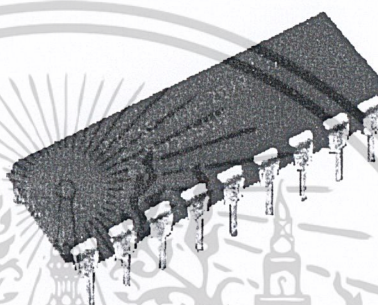
11.ทดสอบการทำงานทั้งหมดของโครงงานเพื่อให้มีการทำงานสัมพันธ์กันระหว่างการควบคุมอุณหภูมิ ความชื้น ที่สภาวะต่างๆกัน พร้อมทั้งบันทึกผล และสรุปผลการทำงานทั้งหมด



บทที่ 2

ทฤษฎี และ หลักการ

2.1 Microcontroller (PIC 16F877)



รูปที่ 2.1 microcontroller ตระกูล PIC

PIC คือ microcontroller อีกตระกูลหนึ่ง ย่อมาจากคำว่า Peripheral Interface Controller ซึ่ง concept ของเจ้า microcontroller ตระกูลนี้ก็คือ พยายามรวมเอาทุกอย่างเอาไว้ในตัวของมันไม่ว่าจะเป็น PROGRAM MEMORY, RAM, EEPROM, SERIAL, I2C, PWM, A/D ฯลฯ โดยไม่จำเป็นต้องต่ออุปกรณ์เสริมจากภายนอกในตัวของ PIC จะมีฟังก์ชันที่ใช้ในการประมวลผล รวมทั้งหน่วยความจำ ซึ่งทำให้มันเหมือนกัน CPU ตัวอื่นๆเลยทีเดียว

2.1.1 ความเร็วของ PIC

ภาคของความถี่สัญญาณนาฬิกา ปัจจุบันสามารถทำสัญญาณนาฬิกาได้ที่ 20 MHz ซึ่งทำให้หนึ่งคำสั่งของ PIC ใช้เวลาเพียง 0.25 uSec แต่อย่างไรก็ตามได้มีบริษัทอื่นได้ซื้อลิขสิทธิ์ PIC จาก microchip และได้สร้าง chip ที่มีความเร็วได้มากกว่าเดิมขึ้นไปอีก

2.1.2 หน่วยความจำของ PIC

ในอดีตหน่วยความจำของ PIC จะค่อนข้างน้อย คืออยู่ระหว่าง 512 words ถึง 4K words แต่ในปัจจุบัน บริษัท microchip ซึ่งเป็นเจ้าของ PIC ได้พัฒนาจนทำให้ memory ของ PIC มีขนาดเป็นหลายสิบกิโลไบต์ และมีที่ว่างจะขยายได้ใหญ่ขึ้นเรื่อยๆ ในเรื่องของการนับขนาดของหน่วยความจำของ PIC จะนับไม่เหมือนปกติ โดยที่หนึ่งคำสั่งของ PIC จะมีขนาด 14 bits ดังนั้นเราจะเรียกว่า 1 word ของ PIC จะมีขนาด 14 bits เช่น PIC16F84A ระบุว่าหน่วยความจำ 1 K (ซึ่งหมายถึง 1 Kword ถ้าคำนวณให้เป็นแบบ 1 byte = 8 bit จะได้ว่า $1 \times 1,024 \times 14 = 14,336$ bits ดังนั้นก็คือ $14,336 / (8 \times 1,024) = 1.75K$ bytes นั่นเอง

2.1.3 สถาปัตยกรรมของ PIC

ตอนนี้มี 3 สายหลักๆ สมัยก่อนมีแค่สอง คือขึ้นต้นด้วย 16xxx, 17xxx และใหม่ล่าสุดคือ 18xxx ถ้าพูดถึง คุณสมบัติที่เหนือกว่าเรียงจากน้อยสุดไปมากที่สุดก็คือ 16 -> 17 -> 18 คำสั่ง assembly ของ 17 และมี 18 จะมีมากกว่า 16 ทำให้เขียนโปรแกรมได้ง่ายกว่า ราคาจะสูงกว่าด้วย แต่ที่เป็นที่นิยมก็คือตระกูล 16xxx ซึ่งโดยทั่วไปแล้ว PIC จะยึดถือการออกแบบที่รวบรวมทุกอย่างไว้ใน chip ตัวเดียวโดยไม่ต้องต่ออุปกรณ์ใดๆ เพิ่มเติม ผลที่ตามมาคือแผ่นวงจรจะมีขนาดเล็ก และอุปกรณ์ที่ใช้จะไม่มาก บางงานอาจจะใช้แค่ PIC เพียงตัวเดียวโดยไม่ต้องใช้ chip อื่นมาเพิ่มเติมเลย นี่ก็คือคุณสมบัติพิเศษของ PIC ซึ่งปัจจุบันหลายบริษัทที่ผลิต Microcontroller ก็เริ่มจะหันมาเลียนแบบแนวทางนี้ แต่ทุกอย่างย่อมมีข้อเสีย เนื่องจาก concept ที่จะรวมทุกอย่างไว้ใน chip เดียว ทำให้ program memory และ data memory ไม่สามารถขยายโดยใช้กับ memory ภายนอกได้ (ในทางทฤษฎี ของจริงนี้ทำได้ แต่ต้องใช้เทคนิคเล็กน้อย ซึ่งไม่นิยม) PIC จึงเหมาะสำหรับงานเล็กๆ ไม่ใช่งานใหญ่ๆ ที่ต้องใช้การคำนวณ และ memory มากมาย

2.1.4 ชนิดของ PIC

MCU ในตระกูล PIC ถิ่นแบ่งออกตามชนิดของ PROGRAM MEMORY แบ่งได้เป็น 3 แบบคือ

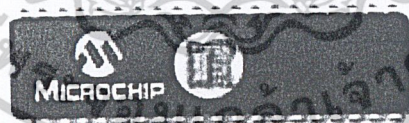
1. OTP (One Time Programmable) OTP เป็น chip ที่มีราคาถูกที่สุด ในสามประเภท สาเหตุก็มาจากว่า chip แบบ OTP จะสามารถทำการโปรแกรมได้แค่ครั้งเดียวเท่านั้น หลังจาก chip ได้ถูกโปรแกรมไปแล้วจะไม่สามารถโปรแกรมเข้าไปใหม่ได้อีก ดังนั้น chip ประเภทนี้จะนิยมใช้หลังจากได้พัฒนาโปรแกรมจนกระทั่งจุดบกพร่องต่างๆในโปรแกรม ไม่มีอีกแล้ว เพราะจะมีต้นทุนต่ำเมื่อเทียบกับ memory ประเภทอื่น จะมีตัวอักษร C แสดงบนตัว chip เช่น 16C84, 16C74



40-LEAD PDIP
"P" OR "PL"

รูปที่ 2.2 OTP (One Time Programmable)

1. EPROM (Erasable Programmable ROM) EPROM เป็น chip ที่มี program memory ที่เมื่อเขียนโปรแกรมเข้าไปแล้วสามารถโปรแกรมใหม่ด้วยการลบโปรแกรมเดิมโดยให้แสง UV (ultra Violet) ส่องผ่านเข้าไปยัง chip ประมาณ 5-10 นาที ดังนั้นที่ด้านบนของ chip จะมีกรอบกระจกเพื่อให้แสง UV สามารถส่องผ่านเข้าไปในตัว chip ได้ แต่ก็มีจำนวนครั้งในการลบโปรแกรมเช่นกัน เมื่อลบโปรแกรมด้วยแสง UV มากๆ ตัว chip ก็จะหมดอายุเร็วขึ้น



40-LEAD CERDIP
"JW"

รูปที่ 2.3 EPROM (Erasable Programmable ROM)

3. EEPROM /Flash (Electrically Erasable Programmable ROM) EEPROM /Flash เป็น chip ที่ออกมาไม่กี่ปีเอง ส่วนของ program memory สามารถอ่านหรือเขียนด้วยสัญญาณทางไฟฟ้า ใช้เวลาในการ ลบข้อมูลไม่กี่วินาที และสามารถลบ และเขียนใหม่ได้หลายพันครั้ง ทำให้เป็นที่นิยมที่สุดใน 3 ประเภท มีตัวอักษร F เป็นตัวบอก เช่น 16F84, 16F877

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



40-LEAD PDIP
"P" OR "PL"

รูปที่ 2.4 EEPROM /Flash (Electrically Erasable Programmable ROM)

2.2 Heater

การเพิ่มอุณหภูมิในโครงงานชุดนี้ ใช้ Heater ในเตาอบไฟฟ้าโดยที่ข้อมูลทางเทคนิคของเตาอบไฟฟ้านี้คือกระแสไฟฟ้า ที่ 220 Volt. ~ 50 Hz. กำลังไฟฟ้า (ฮีตเตอร์) 325 Watt. ขนาดเครื่องภายนอก กว้าง 370 mm. ลึก 235 mm. สูง 219 mm. ความจุ 7 L. ซึ่ง ในการเพิ่มอุณหภูมิใช้ heater ด้านล่างเพียงตัวเดียว เพื่ออำนวยความสะดวก การควบคุมอุณหภูมิ



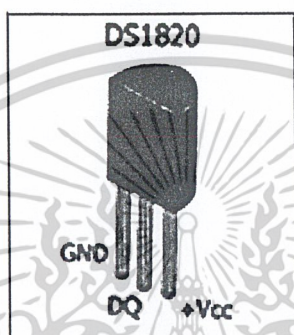
รูปที่ 2.5 เตาอบไฟฟ้า

2.3 ไอซีตรวจจับอุณหภูมิ DS1820

ไอซีตรวจจับอุณหภูมิ DS1820 เป็นไอซีตรวจจับอุณหภูมิที่ใช้การติดต่อแบบระบบบัสหนึ่งสาย มีขาต่อใช้งานเพียง 3 ขา คือ DQ ซึ่งเป็นขาเชื่อมต่อกับระบบบัส, ขาต่อไฟเลี้ยงภายนอก และขากราวด์ ดังแสดงการจัดขาของไอซี DS1820 ในรูปที่ 2.5 หัวใจของ DS1820 อยู่ที่ตัวตรวจจับอุณหภูมิและหน่วยความจำที่มีความเร็วสูงที่เรียกว่า สแครตช์-แพด (scratchpad) ซึ่งมีขนาด 9 ไบต์ เมื่อวัดอุณหภูมิได้ก็นำค่าที่วัดได้มาเก็บไว้ในสแครตช์แพดที่ไบต์ 0 และ 1 ทั้งนี้เนื่องจากไอซี DS1820สามารถให้ข้อมูลของอุณหภูมิได้ละเอียดถึง 16 บิต และเมื่อนำมาแปลงเป็นข้อมูลเลขฐานสิบจึงสามารถแสดงความละเอียดของค่าอุณหภูมิได้ถึง 0.5 องศาเซลเซียสและ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0.9 องศาฟาเรนไฮต์โดยมีย่านวัดอุณหภูมิ -55 ถึง +125 องศาเซลเซียสหรือ -67 ถึง +257 องศาฟาเรนไฮต์ โดยค่าขององศาฟาเรนไฮต์ต้องใช้การแปลงหน่วยเข้ามาช่วย ใช้เวลาในการแปลงค่าอุณหภูมิเป็นข้อมูลดิจิทัลประมาณ 200 มิลลิวินาที สามารถกำหนดขอบเขตของอุณหภูมิที่ทำการวัดได้ และให้แจ้งเตือนเมื่อมีอุณหภูมิสูงขึ้นหรือต่ำลงถึงค่าที่กำหนด โดยค่าอุณหภูมิที่กำหนดจะเก็บไว้ที่สแตนด์บายในไบต์ 2 และ 3



รูปที่ 2.6 การจัดขาของ DS1820

2.4 ตัวทำความชื้นชนิดอัลตราโซนิก (ULTRASONIC HUMIDIFIERS)

ตัวทำความชื้นถูกใช้ในการปรับระดับความชื้น ให้อยู่ในช่วงที่สามารถควบคุมได้ในระบบการทำงานของโรงงานอุตสาหกรรม เช่น การลดอัตราการส่งผ่านพาหะซึ่งเป็นสาเหตุที่ทำให้เกิดเชื้อแบคทีเรียในโรงพยาบาล, การลดไฟฟ้าสถิตย์ในห้องคอมพิวเตอร์และห้องที่ควบคุมเรื่องความสะอาดของโรงงานอุตสาหกรรม

ตัวทำความชื้นชนิดอัลตราโซนิก จะสร้างกลุ่มหมอกไอน้ำที่ปราศจากการเพิ่มขึ้นของอุณหภูมิโดยใช้การสั่นของอุปกรณ์อิเล็กทรอนิกส์ที่ถูกเปลี่ยนให้เป็นการสั่นเชิงกล โดยใช้แผ่นเพียโซ (a piezo disk) จุ่มลงในน้ำ การสั่นเชิงกลจะมีผลโดยตรงที่พื้นผิวน้ำ ซึ่งจะทำให้เกิดความถี่สูงที่บริเวณผิวน้ำทำให้เกิดกลุ่มหมอกของหยดน้ำบริเวณนั้น ขบวนการนี้ไม่สามารถส่งผ่านความร้อนได้ การลดลงของตัวทำความชื้นใช้พลังงานประมาณ 90 ถึง 93 เปอร์เซ็นต์เมื่อเทียบกับระบบที่ทำการต้มน้ำ ตัวทำความชื้นชนิดอัลตราโซนิกแสดงให้เห็นถึงประสิทธิภาพสูงและการซ่อมบำรุงต่ำกว่าตัวทำความชื้นแบบ electrode canisters, quartz lamps and indirect stream-to-stream ขบวนการนี้จะไม่ทำให้เกิดการกัดกร่อนซึ่งส่งผลกระทบต่อคุณภาพของอากาศภายในอาคาร เมื่อเทียบกับตัวทำความชื้นแบบสร้างไอน้ำโดยตรง (direct stream humidifiers)



รูปที่ 2.7 ตัวทำความชื้นชนิดอัลตราโซนิก (ultrasonic humidifiers)

2.5 หน้าจอแสดงผล (LCD DISPLAY)

2.5.1 รายละเอียดเกี่ยวกับโมดูล LCD

ในโมดูล LCD จะมีส่วนประกอบหลักๆ 3 ส่วนดังนี้

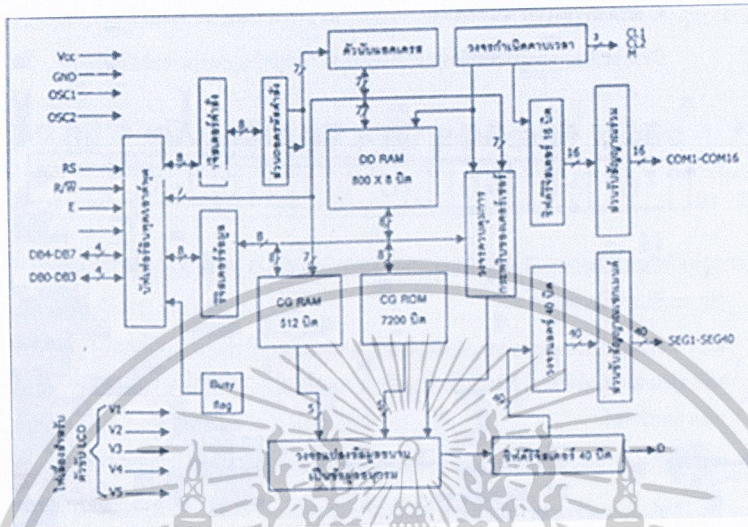
ตัวแสดงผล (display) ภายในเป็นผลึกเหลวที่สามารถแสดงผลให้เห็นโดยอาศัยแสงจากภายนอก ดังนั้นจึงต้องมีมุมในการมองข้อมูลที่แสดงผลบนจอ LCD

ตัวควบคุม (controller) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาควบคุมการทำงานของโมดูล LCD เช่น ลบจอภาพ แสดงตัวอักษร หรือเลื่อนเคอร์เซอร์ เป็นต้น ตัวควบคุมนี้ใช้ชิปควบคุมโดยเฉพาะ ชิปที่นิยมใช้คือ เบอร์ HD44780 และ HD61830 โดย HD44780 จะใช้ควบคุม LCD แบบอักษร ส่วน HD61830 ใช้ควบคุม LCD แบบกราฟิก

ตัวขับ (driver) เป็นตัวรับสัญญาณจากตัวควบคุมมาขับให้ตัวแสดงผลแสดงข้อมูลตามที่กำหนด ชิปที่ใช้ทำหน้าที่เป็นตัวขับนี้ได้แก่ เบอร์ HD44100H และ MSM5259 เป็นต้น

โครงสร้างภายในของตัวควบคุมโมดูล LCD

ในการใช้งานโมดูล LCD จำเป็นต้องทำความเข้าใจเกี่ยวกับโครงสร้างและคำสั่งที่ใช้ในการควบคุมให้ดีเสียก่อน ในหนังสือนี้ขอยกตัวอย่างโมดูล LCD แบบอักษร เพราะสามารถ



รูปที่ 2.8 ไคอะแกรมการทำงานของโมดูล LCD แบบอักษร

บัฟเฟอร์อินพุตเอาต์พุต เป็นส่วนที่ใช้ในการติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอก
 เพื่อที่จะถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม

รีจิสเตอร์คำสั่ง (Instruction Register : DR) เป็นรีจิสเตอร์ใช้รับข้อมูลคำสั่งจาก
อุปกรณ์ภายนอกเพื่อนำไปควบคุมการแสดงผล

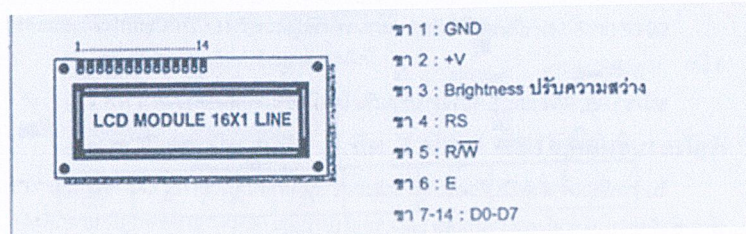
รีจิสเตอร์ข้อมูล (Data Register : DR) เป็นรีจิสเตอร์ที่รับข้อมูลจากอุปกรณ์ภายนอก เพื่อถ่ายทอดต่อไปยังหน่วยความจำที่ทำหน้าที่เก็บข้อมูลแสดงผล หรือนำข้อมูลไปสร้างตัวอักษร ที่เพิ่มเติมในแรมที่เก็บตัวอักษร

แรมเก็บข้อมูลแสดงผล (Display Data RAM : DDRAM) เป็นหน่วยความจำแรมทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์ DR ตัวควบคุมจะนำข้อมูลใน DDRAM นี้ไปเปิดตาราง (Look up-table) ของตัวอักษรที่เก็บไว้ในหน่วยความจำรวมและแรมเก็บตัวอักษร เพื่อนำไปแสดงที่ตัวแสดงผล

รวมเก็บตัวอักษร (Character Generator ROM : CGROM) เป็นหน่วยความจำรวมที่ใช้เก็บข้อมูลตัวอักษรหรือสัญลักษณ์ที่สามารถอ่านออกไปแสดงที่ตัวแสดงผลได้ มีขนาด 7,200 บิต โดยจะถูกอ่านด้วยค่าของข้อมูลใน DDRAM

แรมเก็บตัวอักษร (Character Generator RAM : CGRAM) เป็นหน่วยความจำแรมที่ใช้เก็บอักษรที่มีการสร้างเพิ่มเติมขึ้นใหม่ ในกรณีที่ตัวอักษรใน CGROM ไม่เพียงพอ มีขนาด 512

บิต การเขียนและอ่านค่าไปใช้นั้นทำได้เช่นเดียวกับ CGROM คือ เขียนข้อมูลลงใน DDRAM แล้วตัวควบคุมจะมาอ่านค่าจาก CGRAM เอง



รูปที่ 2.9 รูปร่างและการจัดขาโมดูล LCD แบบอักษร

แฟล็ก BUSY เป็นส่วนที่ทำหน้าที่แจ้งสถานะการทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวควบคุมพร้อมที่จะรับข้อมูลหรือคำสั่งหรือไม่ ดังนั้นก่อนการส่งข้อมูลหรือคำสั่งมายังตัวควบคุมต้องตรวจสอบสถานะของแฟล็ก BUSY นี้เสียก่อน

2.5.2 โมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด (LCD16x1)

สำหรับโมดูล LCD ที่ยกมาใช้ในการเรียนรู้ในการทดลอง เป็นขนาด 16 ตัวอักษร 1 บรรทัด เนื่องจากราคาถูก ง่าย และเป็นโมดูล LCD ที่มีโครงสร้างเป็นมาตรฐาน มีผู้ผลิตหลายราย และมีการระบุเบอร์แตกต่างกันออกไปตามผู้ผลิต อาทิ LM020L ของฮิตาชิ , DMC-16117A ของออพเท็กซ์ (Optrex) เป็นต้น แต่อย่างไรก็ตามคอนโทรลเลอร์ที่ใช้คือเบอร์เดียวกันนั่นคือ เบอร์ HD44750 ของฮิตาชิ

โมดูล LCD ขนาด 16 x 1 มีขาต่อใช้งานทั้งสิ้น 14 ขาดังในรูปที่ 2.7 สำหรับรายละเอียดการทำงานของแต่ละขามีดังนี้

V_{SS} (ขา 1) : ต่อกราวด์

V_{DD} (ขา 2) : ต่อไฟเลี้ยง + 5 โวลต์

V_{OO} (ขา 3) : เป็นขาอินพุตรับแรงดันเพื่อปรับความเข้มของการแสดงผล

RS (ขา 4) : เป็นขาอินพุตใช้ในการแยกชนิดของข้อมูลที่ทำการประมวลผลในขณะนั้นว่าเป็นคำสั่งสำหรับรีจิสเตอร์ IR หรือเป็นข้อมูลสำหรับรีจิสเตอร์ DR โดยถ้าขานี้เป็น “0” ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขานี้เป็น “1” ข้อมูลที่ส่งมาจะเป็นข้อมูลสำหรับการแสดงผล

RW (ขา 5) : เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับ LCD ถ้าเป็น “0” เป็นการกำหนดให้เขียนข้อมูล แต่ถ้าเป็น “1” จะเป็นการอ่านข้อมูล

E (ขา 6) : เป็นขาอินาเบิล LCD ให้ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D₀-D₇ (ขา7-14): เป็นขาที่ใช้เป็นทางผ่านของข้อมูลระหว่าง LCD กับอุปกรณ์ภายนอกขนาด 8 บิต

2.5.3 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD

ในการเขียนข้อมูลเพื่อควบคุมให้โมดูล LCD แสดงผลตามที่ผู้ใช้งานต้องการ ต้องส่งคำสั่ง (Instruction) แล้วกำหนดโหมดการทำงานให้แก่โมดูล LCD ก่อน จากนั้นจึงส่งข้อมูล (data) ที่ต้องการแสดงผลเนื่องจากบัสข้อมูลของโมดูล LCD มี 8 เส้นคือ D0-D7 และใช้เป็นทางผ่านของทั้งคำสั่งและข้อมูล ดังนั้นในการส่งคำสั่งและข้อมูลจึงต้องอาศัยการกำหนดสัญญาณลอจิกที่ขา RS ถ้าหากที่ขา RS ได้ลอจิก “0” หมายความว่า ข้อมูลที่ป้อนให้แก่โมดูล LCD ขณะนั้นเป็นคำสั่งในทางตรงข้าม หากขา RS ได้รับลอจิก “1” ข้อมูลที่ป้อนให้ขณะนั้นเป็นข้อมูลที่ใช้ในการแสดงผล

เมื่อต้องการเขียนหรืออ่านข้อมูลใน CGRAM และ DDRAM เริ่มต้นต้องกำหนดแอดเดรสที่ต้องการอ่านหรือเขียนก่อน โดยใช้คำสั่งเลือกแอดเดรส จากนั้นกำหนดให้ขา RS เป็น “1” เพื่อแจ้งให้ตัวควบคุมภายในโมดูล LCD ทราบว่าข้อมูลที่ปรากฏต่อไปนี้เป็นข้อมูลปกติไม่ใช่คำสั่ง

ในกรณีที่ต้องการอ่านข้อมูลต้องกำหนดให้ขา R/W เป็น “1” ข้อมูลขนาด 8 บิต (หรือ 4 บิต) ก็จะปรากฏบนบัสข้อมูล โดยข้อมูลที่อ่านออกมาได้จะเป็นข้อมูลจากแอดเดรสของ CGRAM หรือ DDRAM ตามที่ต้องการ

ในกรณีที่ต้องการเขียนข้อมูล เมื่อกำหนดแอดเดรสและป้อนลอจิก “1” ให้ขา RS แล้วก็ต้องกำหนดให้ขา R/W เป็น “0” ข้อมูลที่อยู่บนบัสข้อมูลจะถูกเขียนลงในรีจิสเตอร์ DR จากนั้นจึงถ่ายทอดลงใน DDRAM ต่อไป

2.5.4 จังหวะการทำงานของ LCD โมดูล

ในการติดต่อกับโมดูล LCD จะต้องมีการหน่วงเวลาลงหลังจากที่ทำการส่งรหัสคำสั่งหรือข้อมูลเนื่องจากต้องรอให้คอนโทรลเลอร์ภายใน LCD โมดูล แปลความหมายของรหัสคำสั่งและทำงานตามคำสั่งให้เรียบร้อยก่อน จากนั้นจึงจะรับข้อมูลหรือดำเนินการต่อไป

ดังนั้น ในการใช้งานโมดูล LCD ผู้เขียนโปรแกรมต้องมีโปรแกรมเพื่อหน่วงเวลารอให้โมดูล LCD พร้อมทำงานด้วย โดยเมื่อเริ่มจ่ายไฟให้แก่โมดูล LCD ต้องรอประมาณ 10 มิลลิวินาที เพื่อให้โมดูล LCD ทำการเตรียมความพร้อมหรืออินิเชียล (initial) หลังจากนั้นก็จะกำหนดลอจิกให้ขา RS ของโมดูล LCD แล้วต้องหน่วงเวลาอีกประมาณ 2 มิลลิวินาทีเพื่อให้คอนโทรลเลอร์ใน LCD โมดูล แปลความหมายของลอจิกที่ขา RS ว่า ข้อมูลต่อไปที่จะได้รับนั้นเป็นรหัสคำสั่งหรือเป็นข้อมูลที่ต้องการแสดงผล จากนั้นจะเป็นการส่งข้อมูลมารอที่บัสข้อมูล D0-D7 (กรณีทำงานในโหมด 8 บิต) ขั้นตอนต่อไปจะเป็นการส่งสัญญาณพัลส์ไปที่ขา E เพื่ออินิเชียลโมดูล

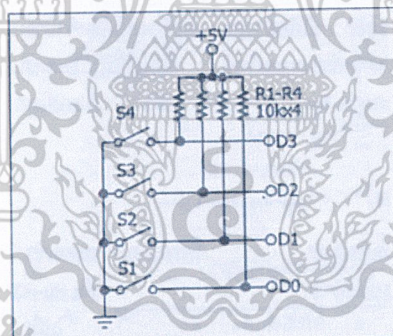
LCD ให้รับข้อมูลจากบัสข้อมูลเข้าไป โดยพัลส์ที่ป้อนเข้าที่ขา E ของ โมดูล LCD ต้องเป็นพัลส์ ขอบขาขึ้น จากนั้นทำการหน่วง เวลา 2 มิลลิวินาที

ทั้งหมดที่กล่าวมาคือขั้นตอนและจังหวะในการทำงาน 1 รอบของโมดูล LCD จะเห็นได้ว่ามีโปรแกรมย่อยที่สำคัญอยู่ 3 โปรแกรมย่อยคือ โปรแกรมอินนิเชียล LCD , โปรแกรมหน่วง เวลา และ โปรแกรมย่อยการส่งพัลส์เพื่ออินา เบิลโมดูล LCD

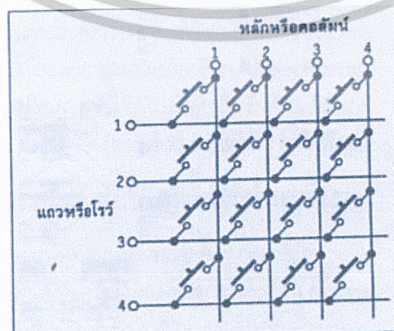
2.6 คีย์แพด หรือ สวิตช์เมตริกซ์ 4 x 3 จุด

การอ่านค่าหรือรับค่าการกดสวิตช์เป็นอีกงานหนึ่งที่ไม่โครคอนโทรลเลอร์ต้องสามารถรองรับและเชื่อมต่อใช้งานร่วมด้วยได้วงจรของสวิตช์มีด้วยกัน 2 ลักษณะใหญ่ๆ คือ ต่อเข้ากับ ไฟเลี้ยงหรือกราวด์โดยตรง เมื่อสวิตช์ตัวใดต่อวงจรสามารถอ่านค่าได้โดยตรง ดังในรูปที่ 2.8 วงจรในลักษณะนี้ไม่มีความซับซ้อน สามารถอ่านค่าของสวิตช์ได้ง่ายและรวดเร็ว แต่มีข้อเสียคือ ถ้าหากจำนวนของสวิตช์มีมากมายจำนวนของสายข้อมูลก็จะมีมากตามทำให้ระบบหรือวงจรโดยรวม มีขนาดใหญ่และสิ้นเปลือง

วงจรของสวิตช์อีกลักษณะหนึ่งคือ การต่อวงจรแบบเมตริกซ์ (matrix switch) ดังในรูปที่ 2.9 สวิตช์จะถูกต่อกันในแนวแกนตั้งและแนวนอน จะเรียกแนวตั้งว่า หลักหรือคอลัมน์ (column)



รูปที่ 2.10 วงจรของสวิตช์แบบต่อเข้ากับไฟเลี้ยงและกราวด์



รูปที่ 2.11 วงจรสวิตช์แบบเมตริกซ์หรือคีย์แพด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขณะที่แนวนอนจะเรียกว่า แถวหรือร็ว (row) ดังนั้นค่าของสวิตช์จะต้องประกอบด้วยตำแหน่งในแนวหลักและแถวกระบวนการที่จะทำให้ได้มาซึ่งค่าของสวิตช์มีขั้นตอนซับซ้อนพอสมควร แต่วงจรของสวิตช์แบบนี้มีข้อดีคือสามารถรองรับการเพิ่มของสวิตช์แบบเมตริกซ์เป็นที่นิยมใช้มากในระบบควบคุมอัตโนมัติหรือกึ่งอัตโนมัติที่มีจำนวนสวิตช์มากกว่า 8 ตัว ในการใช้งานทั่วไปจะเรียกสวิตช์แบบเมตริกซ์นี้ว่า คีย์แพด (keypad)



บทที่ 3

การออกแบบและการสร้าง

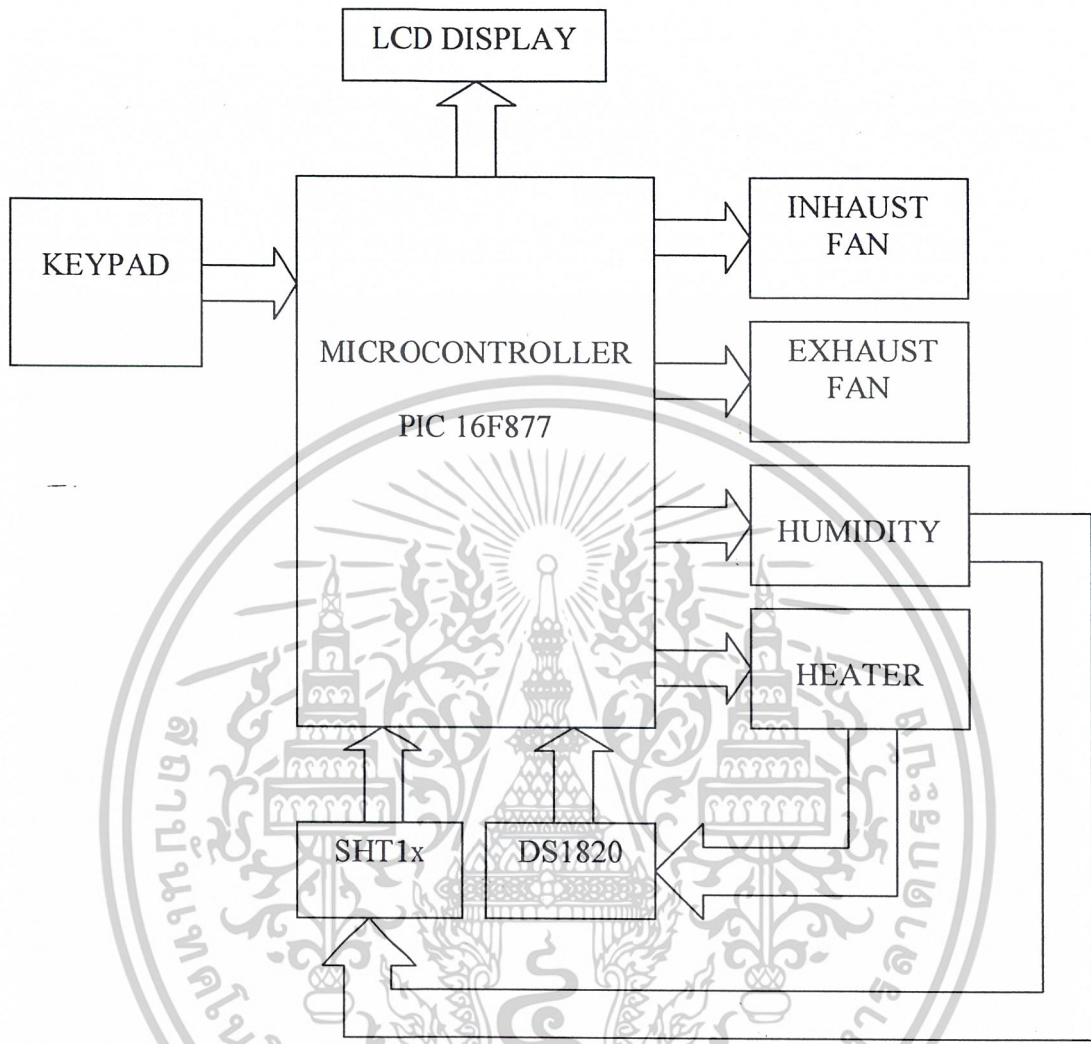
3.1 หลักการทำงาน

หลักการทำงานเริ่มต้นจากการรับค่า(input)ที่ต้องการผ่านทาง คีย์แพด แล้วส่งค่าที่ต้องการไปยัง หน่วยประมวลผล (Microcontroller) และแสดงค่า(output) ผ่านทางหน้าจอแสดงผล (LCD) จากนั้น หน่วยประมวลผล (Microcontroller) ส่งค่าไปควบคุมการทำงานของ พัดลมดูดอากาศ (inhaust fan) พัดลมระบายอากาศ(exhaust fan) ตัวทำความร้อน(heater) ตัวทำความชื้น(humidity) ตัววัดอุณหภูมิ(DS 1820) ทำหน้าที่วัดอุณหภูมิภายในระบบแล้วส่งสัญญาณ ไปยัง หน่วยประมวลผล(Microcontroller) ส่วนตัววัดความชื้น(SHT1x) ทำหน้าที่วัดความชื้นภายในระบบแล้วส่งสัญญาณไปยังหน่วยประมวลผล(Microcontroller) เช่นกัน ดังรูปที่ 3.1

3.2 ส่วนประกอบของโครงงาน

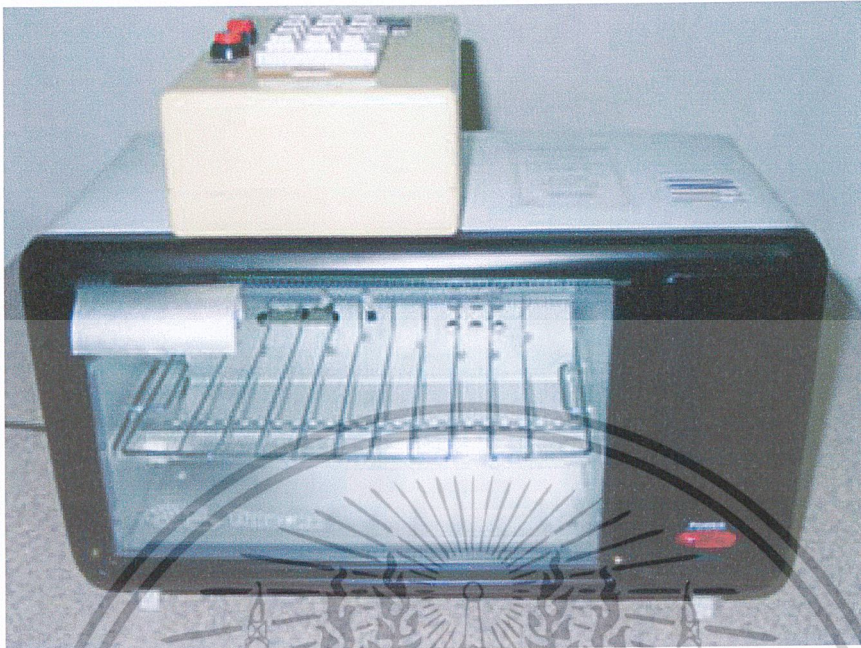
ระบบควบคุมสภาพแวดล้อมในห้องปิด เป็นการควบคุมอุณหภูมิและความชื้นสัมพัทธ์ที่อยู่ในระบบ ดังนั้นจึงต้องประกอบด้วย

1. ส่วนอินพุต เช่น คีย์แพด เซ็นเซอร์วัดสัญญาณ
2. ส่วนประมวลผล ซึ่งเป็นส่วนของโปรแกรมและไมโครคอนโทรลเลอร์
3. หน่วยเอาต์พุต คือ หน้าจอแสดงผล (LCD Module)



รูปที่ 3.1 บล็อกไดอะแกรมของหลักการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงด้านหน้าของโครงการ



รูปที่ 3.3 แสดงด้านข้างของโครงการ

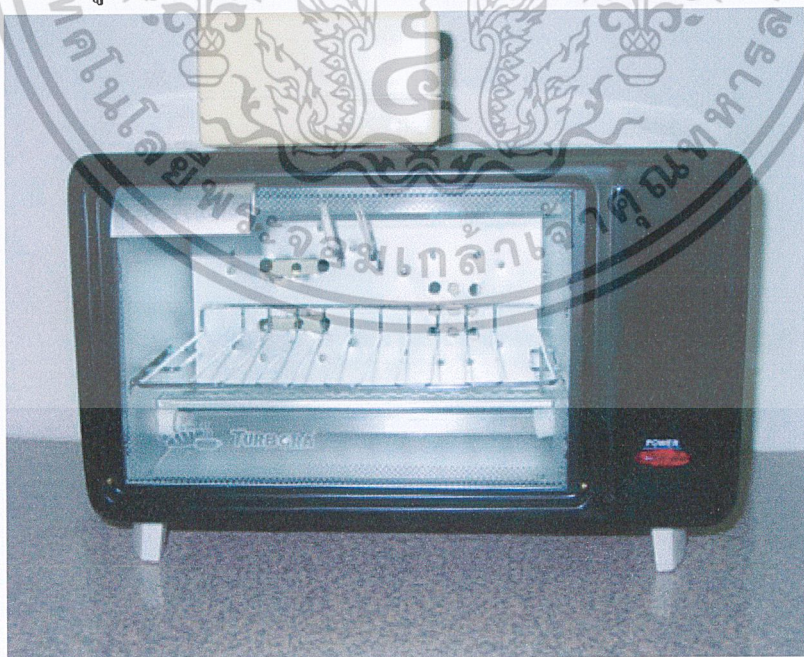
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 แสดงด้านบนของโครงงาน

3.2.1 ห้องปิดจำลอง

ในโครงงานชุดนี้ได้นำเตาอบขนาด 7 ลิตร มีขนาดความกว้าง 23.5 เซนติเมตร ยาว 37 เซนติเมตร สูง 21.9 เซนติเมตร มาใช้เป็นแบบจำลองแทนห้องปิด



รูปที่ 3.5 แสดงด้านหน้าของห้องปิดจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ตัวทำความร้อน (Heater)

ตัวทำความร้อนมีหน้าที่เพิ่มอุณหภูมิให้แก่ระบบ ซึ่งวางอยู่ด้านในของเตาอบ โดยมีกำลังไฟฟ้า 325 วัตต์ ดังรูป



รูปที่ 3.6 แสดงตัวทำความร้อน(coil)

3.2.3 ตัวทำความชื้นชนิดอัลตราโซนิก(Ultrasonic Humidity)

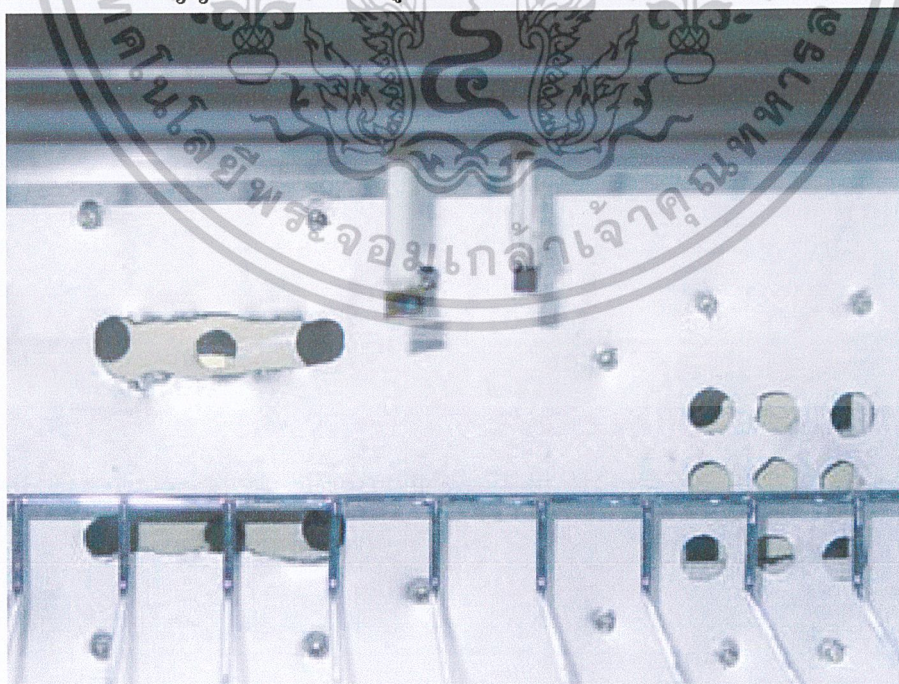
ตัวทำความชื้นมีหน้าที่สร้างความชื้นให้แก่ระบบโดยใช้พัดลมดูดอากาศ (inhaust fan)ดูดความชื้นเข้ามาในระบบ



รูปที่ 3.7 แสดงตัวทำความชื้นชนิดอัลตราโซนิก(Ultrasonic Humidity)

3.2.4 ตัววัดอุณหภูมิ(DS1820) และ ตัววัดความชื้นสัมพัทธ์(SHT1x)

ตัววัดอุณหภูมิมีหน้าที่วัดอุณหภูมิที่อยู่ในระบบและส่งค่าไปประมวลผล ส่วนตัววัดความชื้นก็มีหน้าที่วัดความชื้นที่อยู่ในระบบและส่งค่าไปประมวลผลที่หน่วยประมวลผลเช่นเดียวกัน ซึ่งตัววัดสัญญาณทั้ง 2 จะวางอยู่ตรงกลางของระบบ

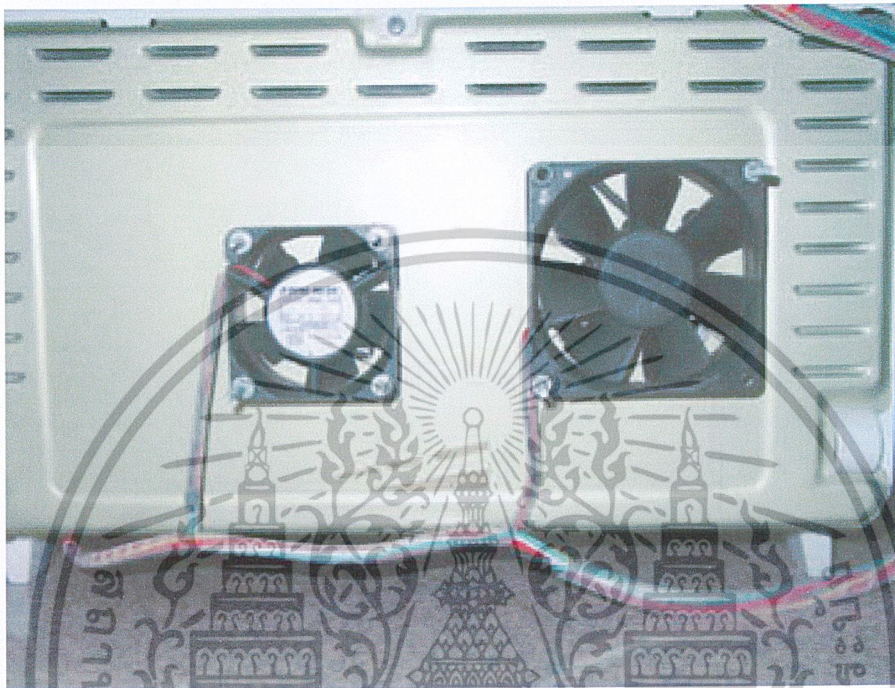


รูปที่ 3.8 แสดงตัววัดอุณหภูมิ(DS1820) และ ตัววัดความชื้นสัมพัทธ์(SHT1x)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 พัดลมระบายอากาศ(exhaust fan) และ พัดลมดูดอากาศ(inhaust fan)

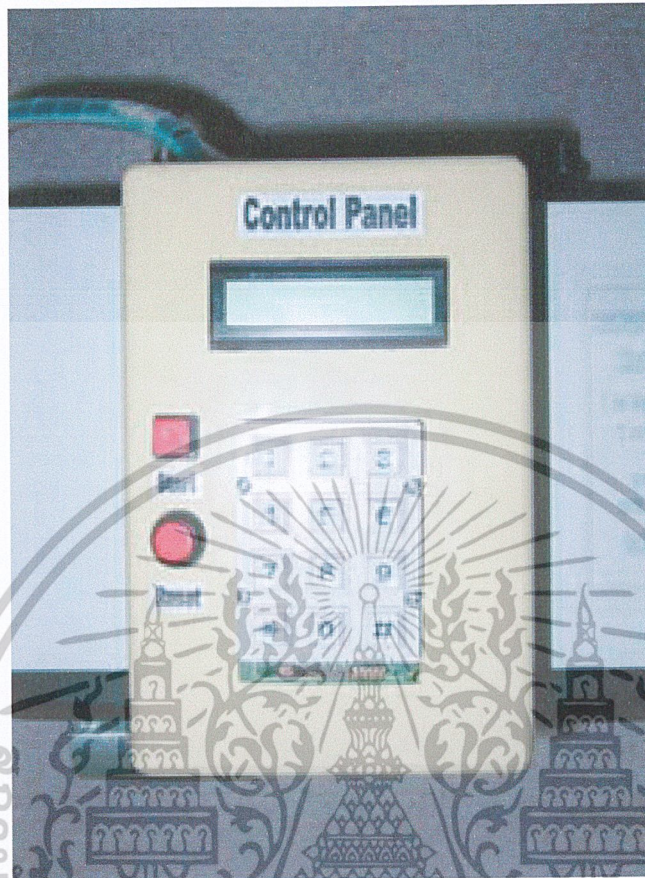
พัดลมระบายอากาศและพัดลมดูดอากาศ ทำหน้าที่ระบายอากาศและดูดอากาศ
สำหรับการลดอุณหภูมิ และเพิ่มความชื้นของระบบ



รูปที่ 3.9 แสดงพัดลมระบายอากาศ(exhaust fan) และ พัดลมดูดอากาศ(inhaust fan)

3.2.6 ส่วนควบคุมและประมวลผล (Control Panal)

เป็นส่วนที่มีหน้าที่รับค่าอินพุต โดยป้อนค่าที่ แผงปุ่มกด(Keypad)และนำค่าไป
ยังหน่วยประมวลผล จากนั้นแสดงค่า เอาท์พุท ที่หน้าจอแสดงผล(LCD Module)



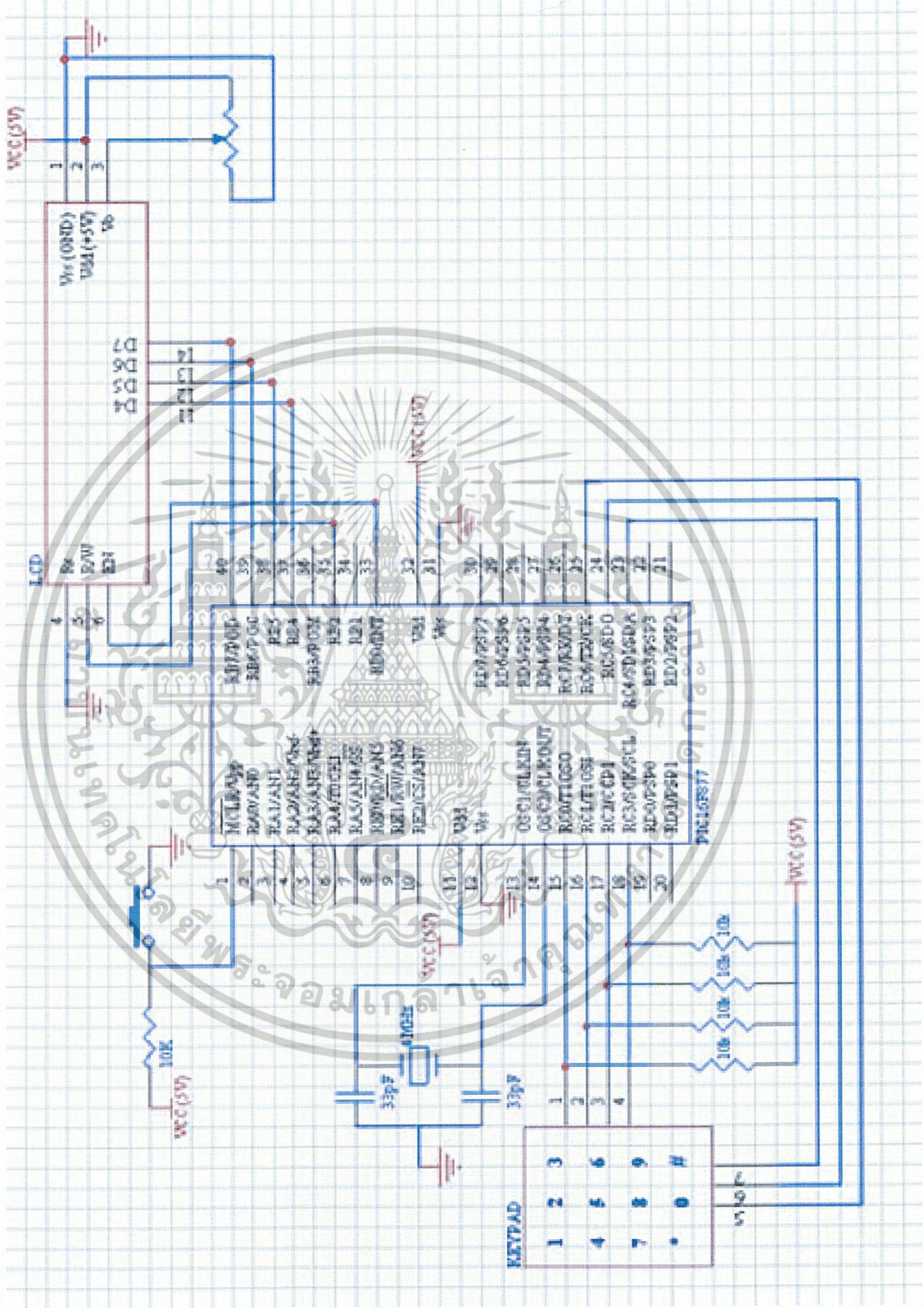
รูปที่ 3.10 แสดงส่วนควบคุมและประมวลผล (Control Panel)

3.3 วงจรควบคุมการทำงานของระบบ

วงจรควบคุมการทำงานของระบบแบ่งออกเป็นส่วนย่อยๆดังนี้

3.3.1 วงจรรับค่าและแสดงผล

จากรูปวงจร แสดงการเชื่อมต่อระหว่าง หน่วยอินพุต(Keypad),หน่วยประมวลผล (Microcontroller)และหน่วยเอาต์พุต(LCD Module)โดยจะทำการรับค่าเข้ามาผ่าน แผง ปุ่มกด ซึ่งจะส่งค่าสัญญาณไปยังตัวประมวลผล และตัวประมวลผลส่งค่าไปยังหน้าจอแสดงผล(LCD Module)

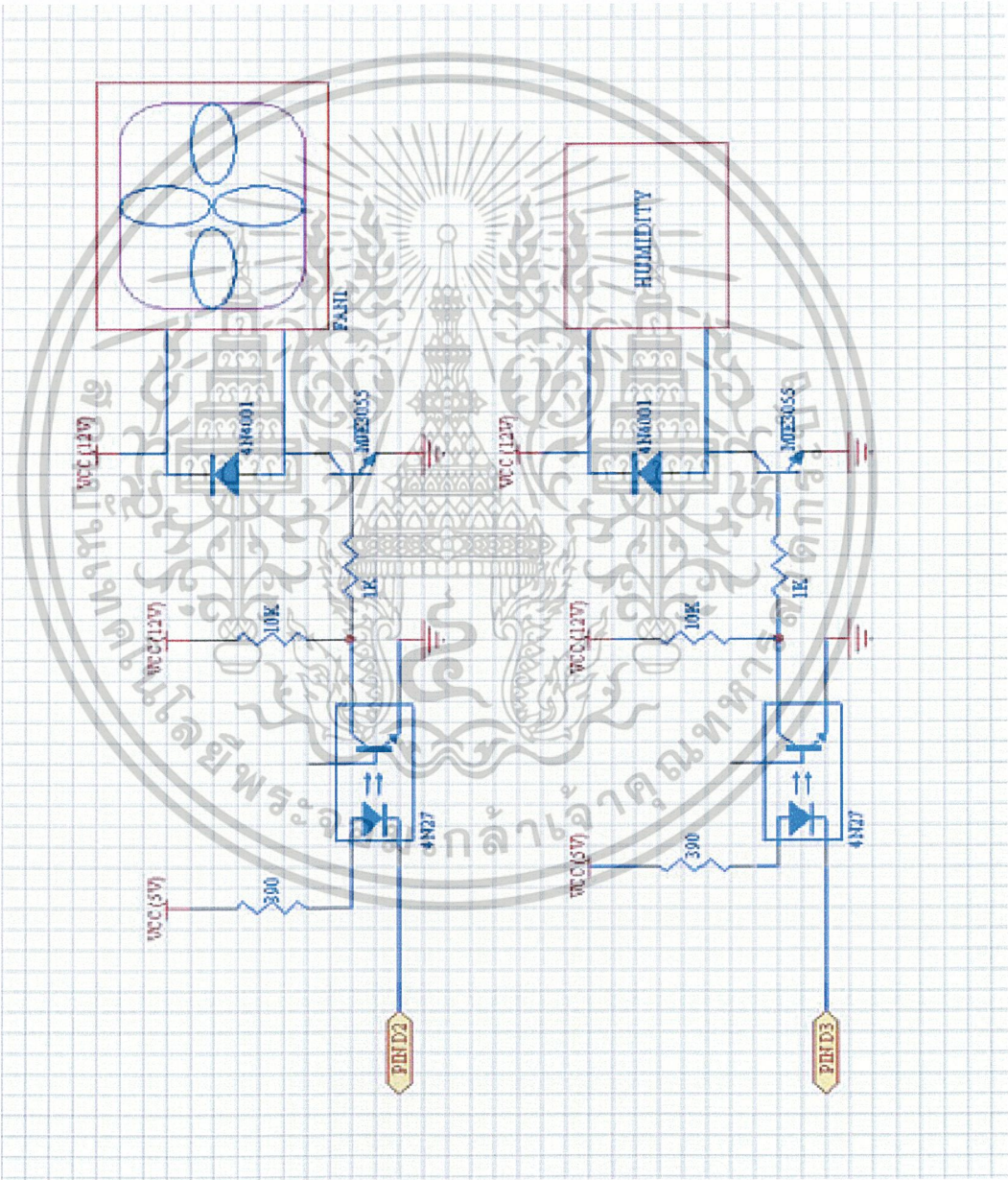


รูปที่3.11แสดงวงจรรับค่าและแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 วงจรควบคุมตัวทำความชื้นและพัดลมระบายอากาศ

เป็นวงจรควบคุมตัวทำความชื้นและพัดลมระบายอากาศ โดยรับสัญญาณจากหน่วยประมวลผลและสั่งให้ทำความชื้นและพัดลมระบายอากาศ ทำงานตามสัญญาณที่ได้รับมา โดยในวงจรจะมีตัว opto.(4N27)ทำหน้าที่แยกกราวด์ ของตัวทำความชื้นและพัดลมระบายอากาศ ออกจาก กราวด์ของของระบบควบคุม



รูปที่ 3.12วงจรควบคุมตัวทำความชื้นและพัดลมระบายอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การควบคุมอุณหภูมิและความชื้น

3.4.1 การควบคุมอุณหภูมิ

ในการควบคุมอุณหภูมิจะใช้ตัวทำความร้อน(heater)เป็นตัวเพิ่มอุณหภูมิ และใช้พัดลม ดูดอากาศและพัดลมระบายอากาศเป็นตัวลดอุณหภูมิ ซึ่งในการควบคุมการเพิ่มอุณหภูมินั้น หากตัวทำความร้อนให้พลังงานความร้อนมาก ก็จะทำให้ควบคุมอุณหภูมิได้ยาก ดังนั้นจึงต้องมีการลดแรงดันเพื่อควบคุมพลังงานความร้อนที่เกิดจากตัวทำความร้อนส่วนในการลดอุณหภูมิจะต้องมีการควบคุมระดับความเร็วของพัดลมเพื่อให้ควบคุมอุณหภูมิได้ง่ายขึ้น

3.4.2 การควบคุมความชื้น

การเพิ่มความชื้นในห้องปิดสามารถทำได้หลายวิธี แต่ในโครงการนี้เลือกใช้ตัวทำความชื้นแบบอัลตราโซนิคมาควบคุมแบบ on-off control และมีการควบคุมพัดลมดูดอากาศเพื่อให้ดูความชื้น เข้ามาในระบบดังรูป



รูปที่ 3.14 แสดงการควบคุมความชื้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

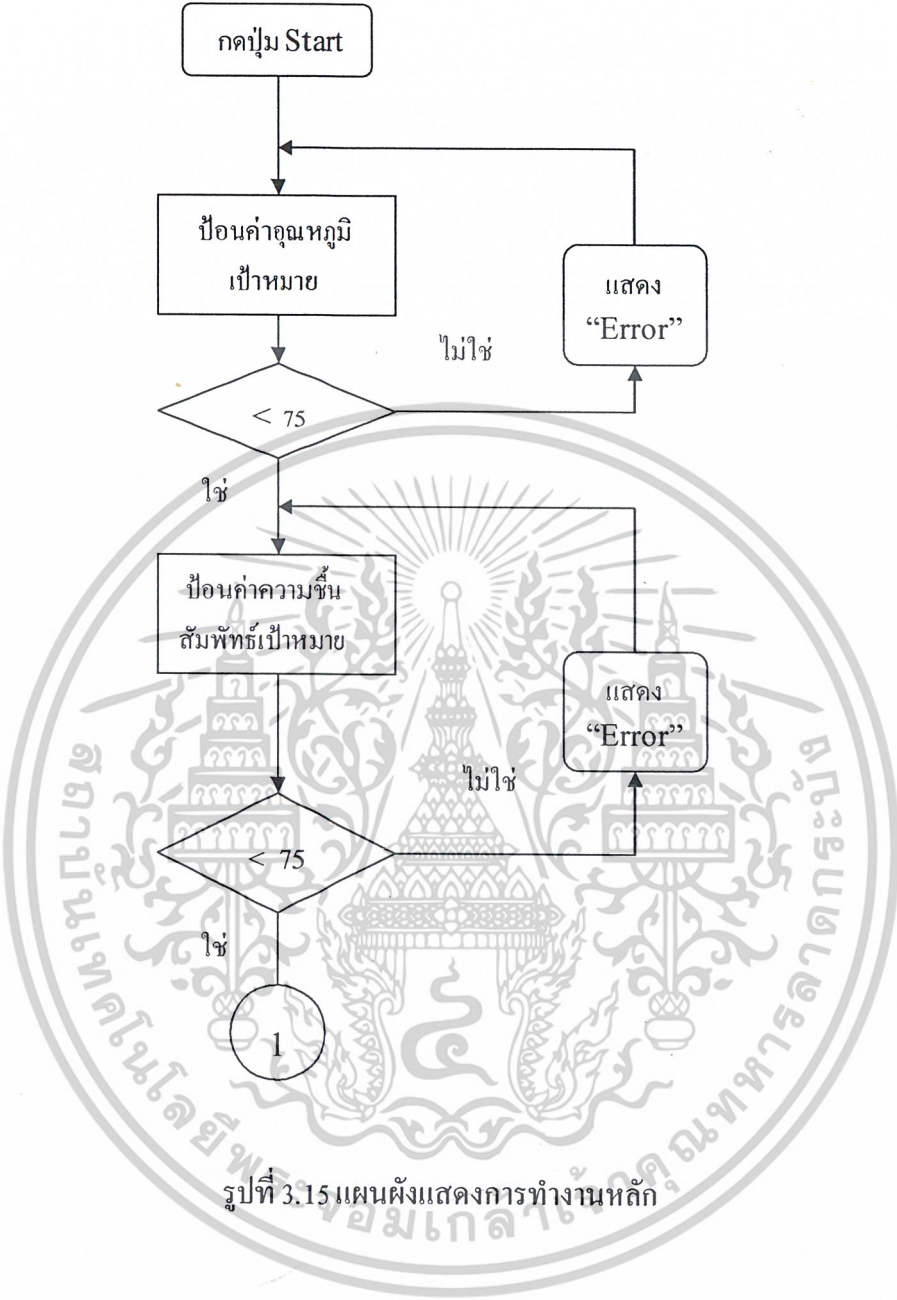
จากรูป เป็นการสร้างความชื้น โดยนำตัวทำความชื้นมาจุ่มลงในน้ำ เพื่อสร้างไอน้ำแล้วใช้พัดลมดูดอากาศ ดูดไอน้ำเข้าไปเพื่อเพิ่มความชื้นให้กับระบบ

3.5 โปรแกรมควบคุมการทำงาน

3.5.1 โปรแกรมหลัก

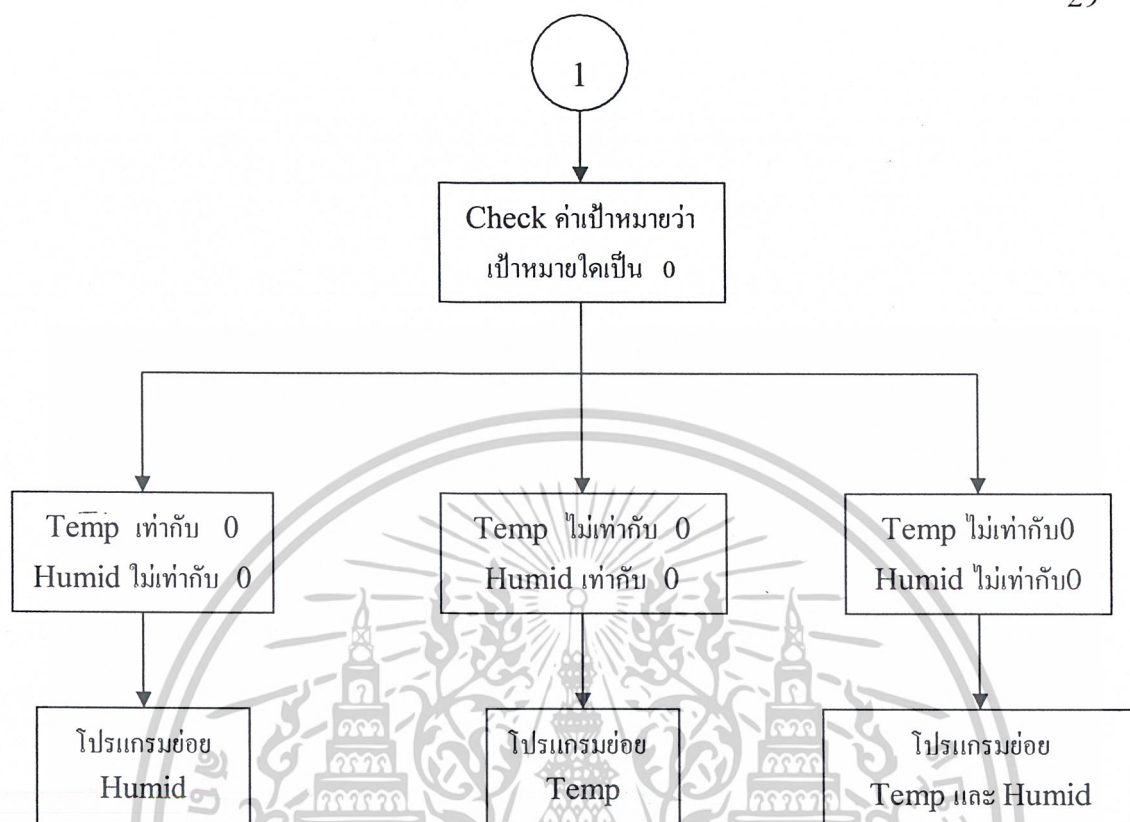
โปรแกรมหลักถือเป็นส่วนสำคัญมากส่วนหนึ่งเพราะโปรแกรมหลักจะคอยจัดการ ตรวจสอบค่าของกระบวนการต่างๆ โดยจะมีการเรียกโปรแกรมย่อยต่างๆ ให้มาทำงานตามการร้องขอของโปรแกรมหลัก

เริ่มต้นการทำงาน โปรแกรมหลักจะแสดงข้อความทางจอ LCD เพื่อให้ทำการป้อนข้อมูลค่าอุณหภูมิและความชื้นที่ต้องการควบคุมให้ได้ตามต้องการ จากนั้นจะเข้าสู่ส่วนของการควบคุมอุณหภูมิและความชื้นให้ได้ตามเป้าหมายหลังจากนั้น โปรแกรมก็จะวนกลับสู่จุดเริ่มต้นใหม่เป็นเช่นนี้ไปเรื่อยๆ จนสามารถควบคุมอุณหภูมิและความชื้นให้ได้ค่าตามต้องการ พร้อมกันนั้นโปรแกรมหลักจะทำการเรียกโปรแกรมตรวจสอบค่าอุณหภูมิและความชื้นภายในระบบและทำการแสดงผลค่าอุณหภูมิ และความชื้นภายในระบบขณะนั้นตลอดเวลาการควบคุมและจะเริ่มต้นการทำงานใหม่เมื่อมีการกดปุ่มรีเซ็ตการทำงาน



รูปที่ 3.15 แผนผังแสดงการทำงานหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

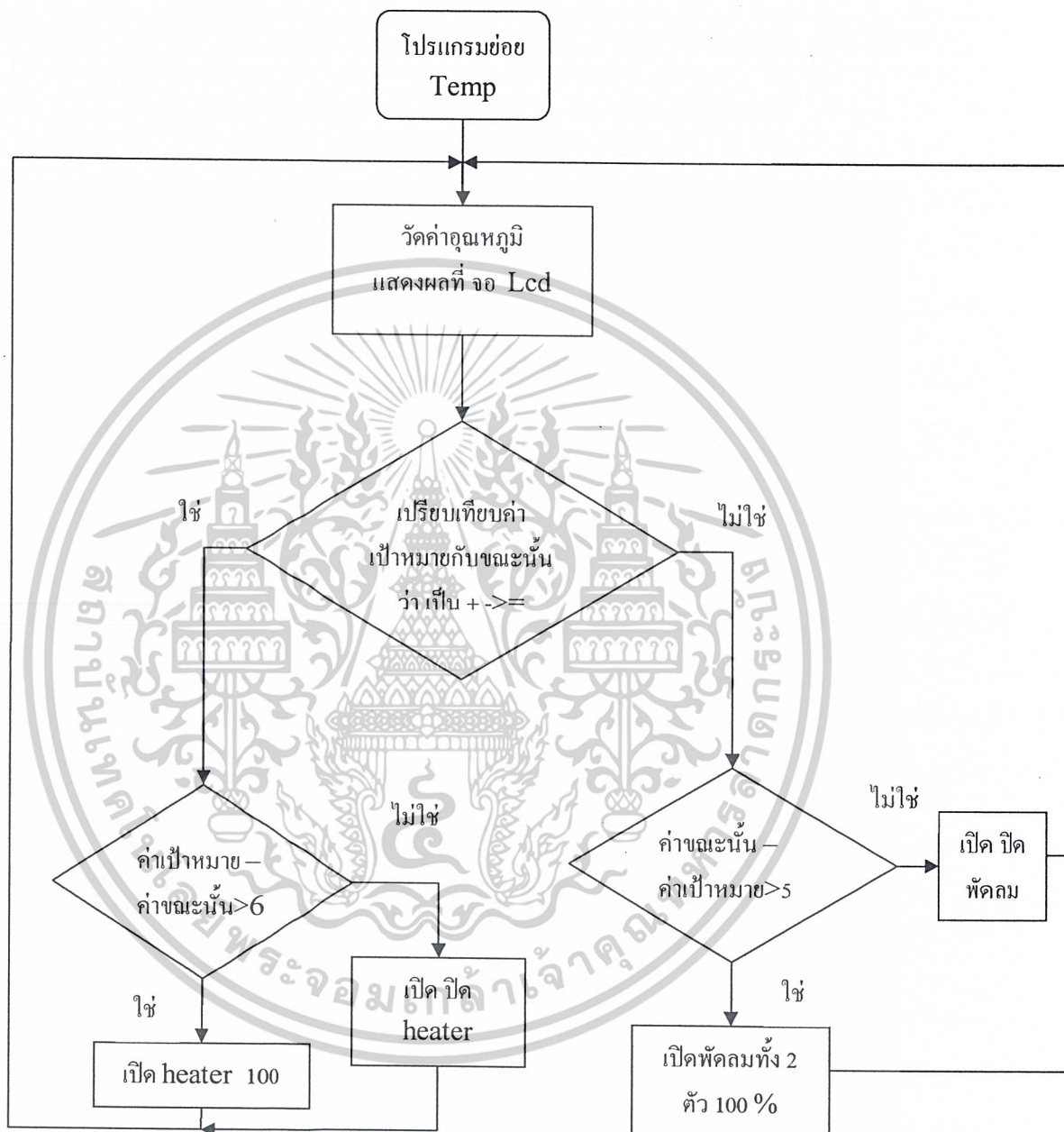


รูปที่ 3.15 (ต่อ) แผนผังแสดงการทำงานหลัก

3.5.2 โปรแกรมย่อยควบคุมอุณหภูมิ

เป็นโปรแกรมหนึ่งซึ่งอยู่ในส่วนของโปรแกรมหลัก โปรแกรมส่วนนี้จะทำการตรวจสอบค่าอุณหภูมิภายในระบบ และทำการเปรียบเทียบกับค่าอุณหภูมิที่กำหนดไว้ เมื่อค่าอุณหภูมิ ในระบบและอุณหภูมิเป้าหมายมีค่าไม่เท่ากัน โปรแกรมจะทำงานเปรียบเทียบค่าทั้งสองว่าค่าอุณหภูมิของระบบสูงกว่าเป้าหมายเท่าไรและทำการสั่งให้พัดลมหมุนด้วยความเร็วต่าง ๆ กัน เพื่อระบายอากาศภายในออกมา และ เป่าอากาศภายนอกเข้ามาในระบบเพื่อลดอุณหภูมิภายในระบบ พร้อมทำการตรวจสอบอุณหภูมิภายในระบบตลอดเวลาเพื่อเปรียบเทียบกับอุณหภูมิเป้าหมายหากยังไม่ได้ค่าอุณหภูมิเป้าหมายก็ทำการสั่งพัดลมไปเรื่อยๆจนได้ค่าตามเป้า แต่หากอุณหภูมิภายในต่ำกว่าเป้าหมาย โปรแกรมจะทำการสั่ง heater ให้ทำงาน ในแรงดันเฉลี่ย ที่ต่างกันเพื่อเพิ่มอุณหภูมิในระบบ โปรแกรมจะทำการเปรียบเทียบค่าอุณหภูมิภายในระบบและอุณหภูมิเป้าหมายและหาเงื่อนไขวนซ้ำกันเพื่อให้อุณหภูมิภายในเข้าสู่ค่าอุณหภูมิเป้าหมายตลอดเวลาจนกว่าจะมีการเช็คค่า

อุณหภูมิเป้าหมายใหม่อีกครั้งโฟลว์ชาร์ตแสดงการทำงานของโปรแกรมควบคุม
อุณหภูมิแสดงดังรูปที่ 3.16

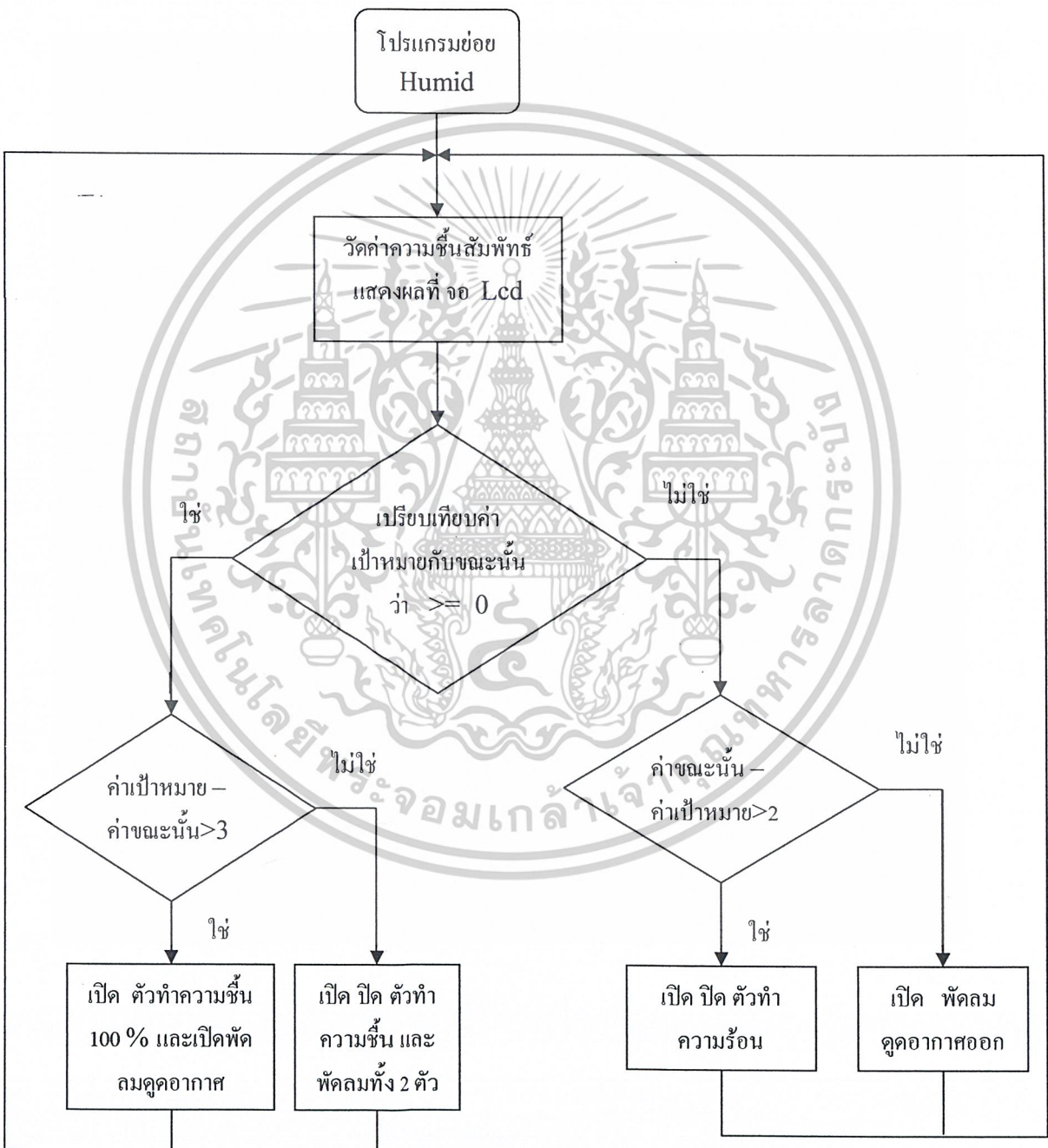


รูปที่ 3.16 แผนผังแสดงการควบคุมอุณหภูมิ

3.5.3 โปรแกรมย่อยควบคุมความชื้น

โปรแกรมนี้อาจจะทำการควบคุมความชื้นให้ได้ค่าความชื้นที่ต้องการโดยจะทำการวัดความชื้นภายในระบบเปรียบเทียบกับค่าความชื้นเป้าหมายว่าได้ค่าตรงกันหรือไม่ หากได้ค่าตรงกันหรือความชื้นในระบบมากกว่าก็จะทำการหยุดการทำงานของเครื่องทำไอน้ำ

แต่หากค่าความชื้นภายในระบบต่ำกว่าค่าเป้าหมายก็ทำการเปิดเครื่องทำไอน้ำเพื่อเพิ่มความชื้นในอากาศและโปรแกรมจะทำการตรวจสอบค่าความชื้นภายในระบบตลอดเวลาเพื่อเปรียบเทียบกับค่าเป้าหมาย จนกว่าจะมีการเซตค่าความชื้นที่ต้องการใหม่อีกครั้ง
โฟลว์ชาร์ตแสดงการทำงานของโปรแกรมควบคุมความชื้นแสดงดังรูปที่ 3.17



รูปที่ 3.17 แผนผังแสดงการควบคุมความชื้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

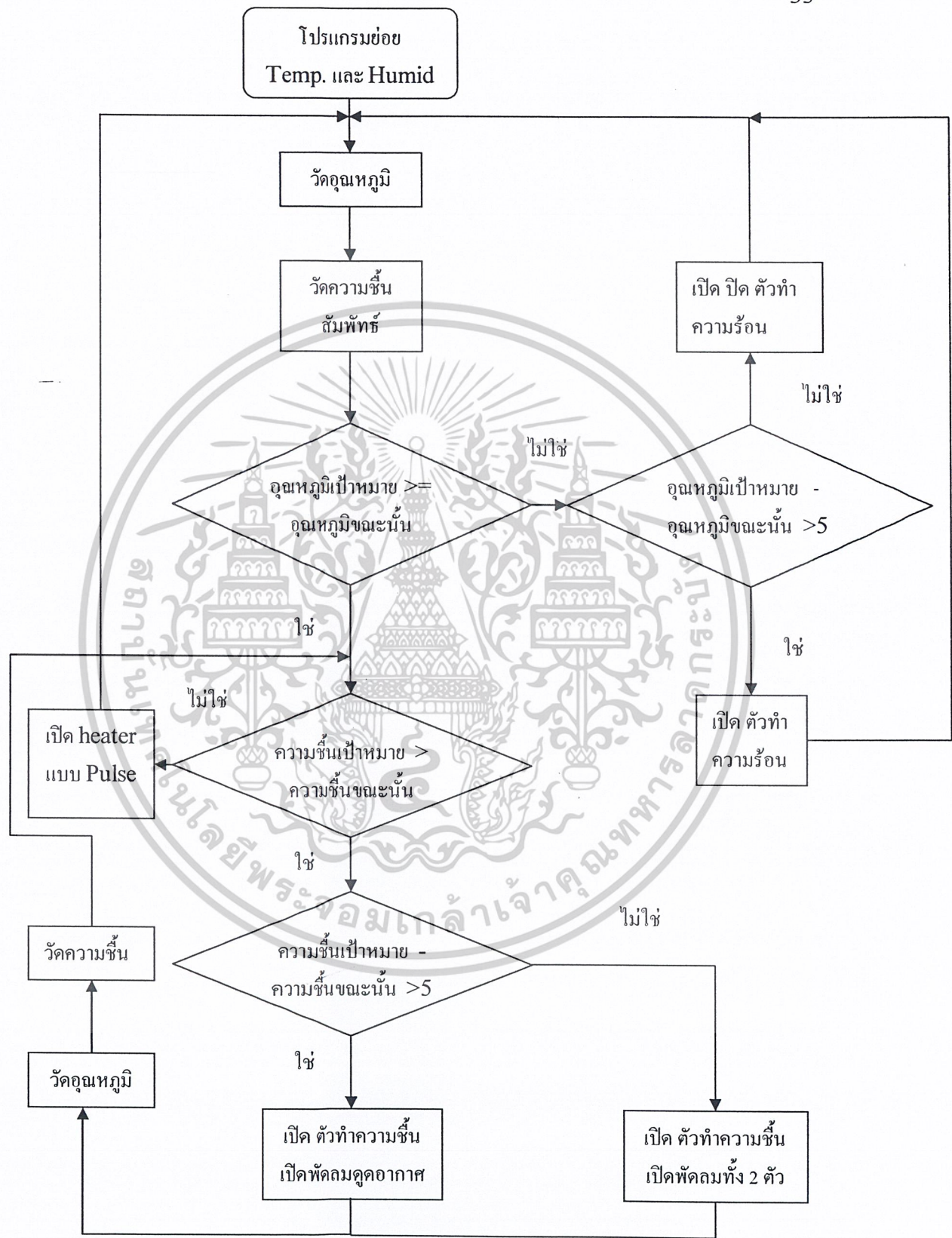
3.5.4 โปรแกรมย่อยควบคุมอุณหภูมิและความชื้น

โปรแกรมนี้จะทำการควบคุมอุณหภูมิและความชื้นไป ด้วยกันและทำการวัดค่าอุณหภูมิและความชื้นตลอดเวลา โดยจะทำการเปรียบเทียบอุณหภูมิเป้าหมายกับอุณหภูมิที่ได้เซ็นค่าไว้ และทำการควบคุมอุณหภูมิให้ได้ตามเป้าหมายก่อน เมื่ออุณหภูมิภายในระบบเท่ากับอุณหภูมิเป้าหมายแล้ว ก็จะมีการไปควบคุมความชื้นให้ได้ค่าตามความชื้นที่ได้ตั้งเป้าหมายไว้เมื่อ ค่าความชื้นได้ตามเป้าหมายก็จะกลับไปควบคุมอุณหภูมิใหม่อีกครั้ง และจะทำการเปรียบเทียบอย่างนี้ต่อไปเรื่อยๆ จนกว่าจะมีการเซ็นค่าเป้าหมายใหม่

ไฟล์ชาร์ตแสดงการทำงานของโปรแกรมควบคุมอุณหภูมิและความชื้นแสดงดัง

— รูปที่ 3.18





รูปที่ 3.18 แผนผังแสดงการควบคุมอุณหภูมิและความชื้นสัมพัทธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การใช้งานโปรแกรมควบคุม

ระบบควบคุมจะประกอบด้วยส่วนสำคัญ 4 ส่วนคือส่วนประมวลผล ส่วนแสดงผล ส่วนติดต่อผู้ใช้ อุปกรณ์ควบคุมขั้นสุดท้าย ซึ่งเซ็นเซอร์และสัญญาณควบคุม โปรแกรมจะทำการเชื่อมโยงส่วนต่างๆ เหล่านี้มาทำงานร่วมกันการทำงานของโปรแกรมจะเป็นไปอย่างอัตโนมัติ และเพื่อประโยชน์สูงสุดโปรแกรมควรจะสามารถปรับเปลี่ยนการทำงานให้ตรงกับผู้ใช้งาน โปรแกรมควบคุมของโครงการนี้ถูกออกแบบมาให้ผู้ใช้สามารถ ปรับเปลี่ยนค่าเป้าหมายต่างๆ ได้ โดยผู้ใช้สามารถติดต่อกับหน่วยประมวลผลผ่านทางคีย์แพดและจอแสดงผล LCD



รูปที่ 3.19 แสดงส่วนติดต่อกับผู้ใช้งาน

จากรูปข้างต้นแสดงถึงส่วนประกอบของส่วนติดต่อผู้ใช้งานซึ่งสามารถอธิบายหน้าที่ได้ดังนี้

- จอแสดงผล จะเป็นจอภาพแบบ LCD ที่จะแสดงตัวอักษร สัญลักษณ์ต่างๆ เพื่อสื่อสารกับผู้ใช้หน่วยประมวลผล
- คีย์แพด เป็นส่วนรับข้อมูลต่างๆจากผู้เข้าไปสู่หน่วยประมวลผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปุ่มกดบน Control Panel สามารถแยกหน้าที่การทำงานได้ดังนี้

- ปุ่มหมายเลข “ 0 – 9 ” ใช้เพื่อป้อนข้อมูลเป้าหมาย
- ปุ่ม “ # ” ใช้เพื่อยืนยันข้อมูลเป้าหมาย
- ปุ่ม “ Start ” ใช้สำหรับเปิด-ปิด การทำงาน
- ปุ่ม “ Reset ” ใช้สำหรับการเปลี่ยนแปลงข้อมูลเป้าหมาย

3.7 ลำดับขั้นตอนการใช้งาน

3.7.1 จ่ายไฟฟ้ากระแสสลับให้แก่เตาอบ และ จ่ายไฟฟ้ากระแสตรงให้แก่ระบบ

3.7.2 กดปุ่ม “ Start ” เพื่อเริ่มต้นการใช้งาน แล้วหน้าจอจะปรากฏข้อความ “Enter Temp.”

3.7.3 ป้อนค่าอุณหภูมิเป้าหมาย แล้วกดเครื่องหมาย “ # ” แล้วหน้าจอจะปรากฏข้อความ “Enter Humid”

3.7.4 ป้อนค่าความชื้นเป้าหมาย แล้วกดเครื่องหมาย “ # ” เพื่อเริ่มต้นการทำงาน

3.7.5 หากมีการผิดพลาดในการป้อนข้อมูลเป้าหมาย ให้กดปุ่ม “ Reset ” เพื่อกลับไปป้อนข้อมูลเป้าหมายใหม่

3.7.6 เมื่อป้อนข้อมูลอุณหภูมิและความชื้นเรียบร้อยแล้วที่หน้าจอจะปรากฏค่าเป้าหมายและค่า อุณหภูมิ,ความชื้นภายในระบบขณะนั้นและจะเปลี่ยนค่าไปเรื่อยๆ จนกระทั่งเข้าสู่ค่าเป้าหมายหลังจากนั้นจึงกดปุ่ม “ Reset ” เพื่อป้อนข้อมูลเป้าหมายใหม่เพื่อให้ระบบทำงานอีกครั้ง

บทที่ 4

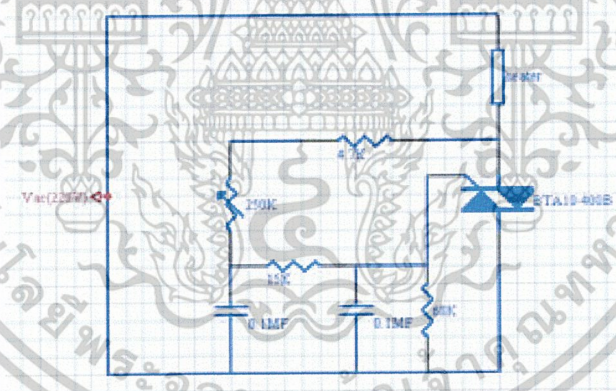
การทดลองและผลการทดลอง

จากที่ได้ทำการออกแบบและจัดสร้าง วงจรระบบควบคุมอัตโนมัติ วงจรตรวจจับอุณหภูมิและตรวจจับความชื้นเมื่อนำมาต่อเข้ากับห้องปิดจำลอง จากนั้นนำมาทำการทดลองควบคุมอุณหภูมิและความชื้น ว่าสามารถควบคุมได้มีประสิทธิภาพเท่าไร

4.1 การทดลองที่ 1 การหาประสิทธิภาพของตัวทำความร้อนที่อยู่ในเตาอบ(heater)

ในการทดลองนี้ได้ทำการจ่ายไฟฟ้ากระแสสลับที่พิกัดแรงดันต่างๆกันให้กับ ตัวทำความร้อน (heater) แล้ววัดอุณหภูมิซึ่งมีการ กำหนดอุณหภูมิเป้าหมายไว้ที่ 70 องศาเซลเซียส โดยมีการทดลองป้อนแรงดันที่ 3 ระดับ ดังนี้คือ

- 1. ป้อนแรงดันที่ 100% ของแรงดันตกคร่อมที่ตัวทำความร้อน
- 2. ป้อนแรงดันที่ 75 % ของแรงดันตกคร่อมที่ตัวทำความร้อน
- 3. ป้อนแรงดันที่ 50 % ของแรงดันตกคร่อมที่ตัวทำความร้อน



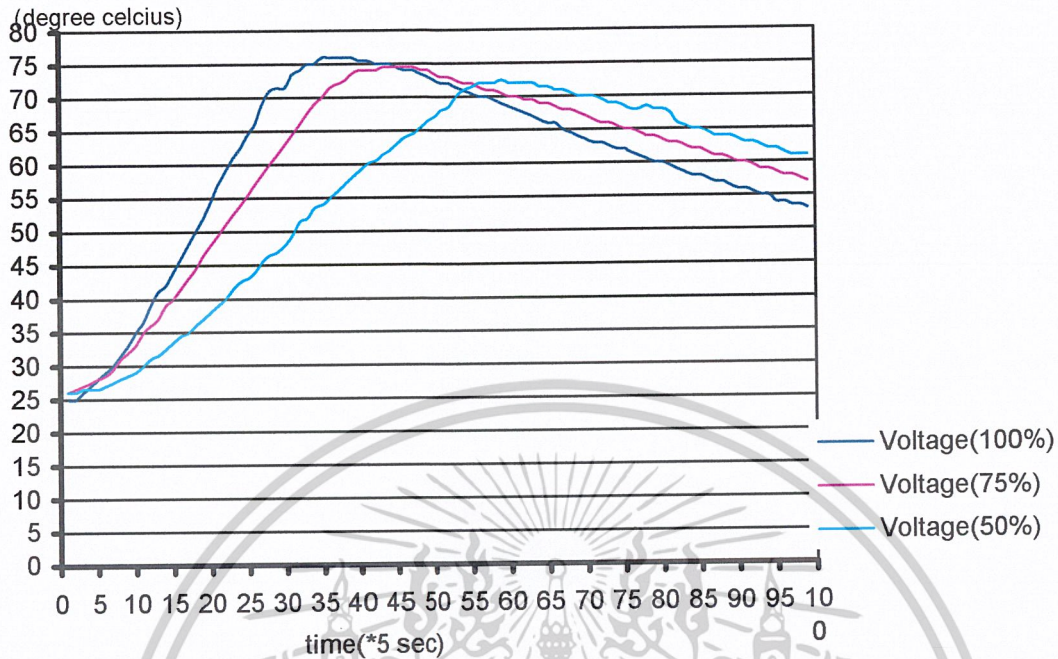
รูปที่ 4.1.1 แสดงวงจรตัวหรีไฟ

ในการกำหนดพิกัดแรงดันที่แตกต่างกัน ได้มีการปรับค่าความต้านทานปรับค่าได้ ของตัวหรีไฟซึ่งมีผลทำให้แรงดันตกคร่อมตัวทำความร้อน เปลี่ยนแปลงตามพิกัดที่ต้องการ

4.1.1 ขั้นตอนการทดลอง

- 4.1.1.1 จ่ายไฟฟ้ากระแสสลับและป้อนพิกัดแรงดันที่ต้องการให้กับเตาอบ
- 4.1.1.2 วัดอุณหภูมิภายในเตาอบ
- 4.1.1.3 บันทึกผลการทดลอง
- 4.1.1.4 นำผลการทดลองที่ได้ไป plot กราฟ แสดงดังรูปที่ 4.1.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1.2 แสดงผลการทดลองหาประสิทธิภาพของตัวทำความร้อน (heater)

4.1.2 สรุปผลการทดลอง

เมื่อมีการจ่ายไฟฟ้ากระแสสลับที่พิกัดแรงดันแตกต่างกันจะมีผลต่อการเพิ่มอุณหภูมิซึ่งแสดงได้ดังรูปที่ 4.1

4.1.2.1. เมื่อป้อนพิกัดแรงดัน 100% ของแรงดันตกคร่อมที่ตัวทำความร้อน การเพิ่มอุณหภูมิจะทำให้เร็วมากซึ่งจะได้ค่า rise time น้อยที่สุด แต่จะให้ค่า Over shoot สูงมาก และเมื่อหลังจากที่อุณหภูมิผ่านค่าอุณหภูมิเป้าหมายไปแล้วค่าอุณหภูมิจะลดลงเร็วมากเช่นกัน

4.1.2.2. เมื่อป้อนพิกัดแรงดันที่ 75 % ของแรงดันตกคร่อมที่ตัวทำความร้อน การเพิ่มอุณหภูมิจะทำให้ได้ไม่เร็วมากเมื่อเทียบกับการป้อนพิกัดแรงดันที่ 100 % แต่จะให้ Over shoot ที่น้อยกว่า และการลดลงของอุณหภูมิหลังจากผ่านค่าอุณหภูมิเป้าหมายมีการลดลงที่ช้ากว่าซึ่งจะสามารถทำให้ควบคุมอุณหภูมิได้ดีกว่า

4.1.2.3. เมื่อป้อนพิกัดแรงดันที่ 50 % ของแรงดันตกคร่อมที่ตัวทำความร้อน การเพิ่มอุณหภูมิจะทำให้ช้าที่สุดซึ่งจะได้ค่า rise time มากที่สุด และจะให้ค่า Over shoot ต่ำที่สุด แต่ไม่เป็นผลดีต่อการควบคุมอุณหภูมิ เนื่องจากค่า rise time มาก ทำให้ควบคุมการเพิ่มอุณหภูมิได้ช้า

4.2. การทดลองที่ 2 การควบคุมอุณหภูมิ ที่ 70 องศาเซลเซียสโดยป้อนพิกัดแรงดันที่ แตกต่าง กัน

ในการทดลองนี้ จะจ่ายไฟฟ้ากระแสสลับที่พิกัดแรงดันที่ แตกต่างกันเหมือนการทดลองที่

4.1 แต่จะทำการป้อน อุณหภูมิเป้าหมายที่ 70 องศาเซลเซียสแล้วทำการควบคุมอุณหภูมิ

4.2.1. ขั้นตอนการทดลอง

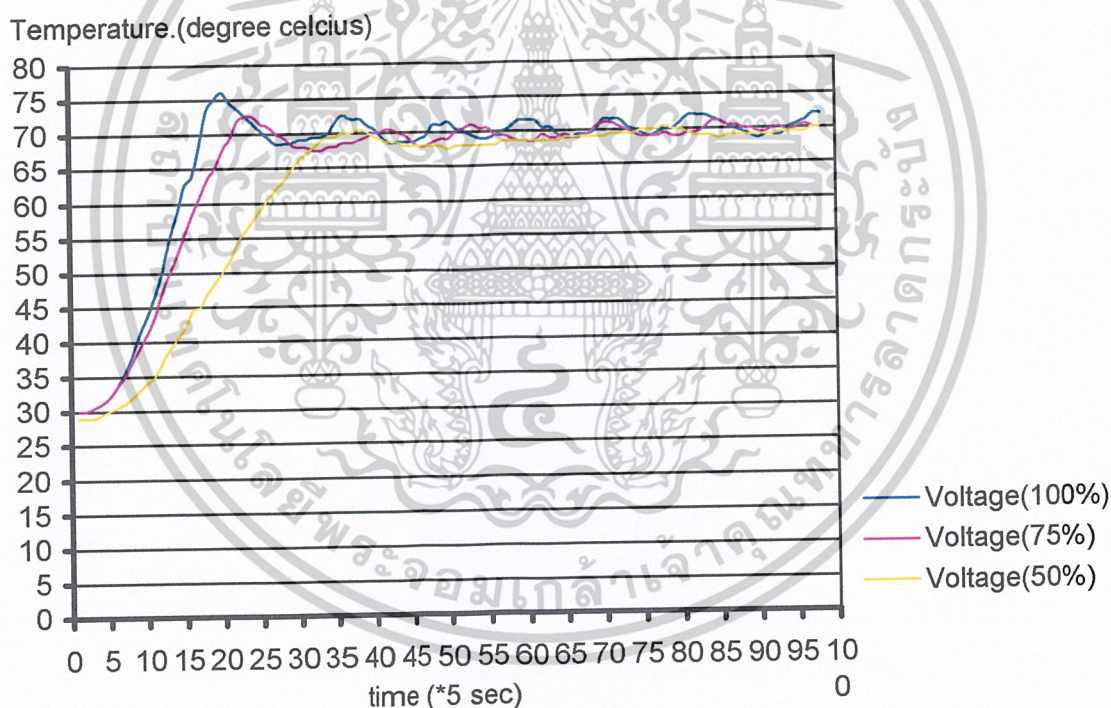
4.2.1.1จ่ายไฟฟ้ากระแสสลับและป้อนพิกัดแรงดันที่ต้องการให้กับเตาอบ และจ่ายไฟฟ้า

กระแสตรงให้กับระบบ

4.2.1.2ป้อนอุณหภูมิเป้าหมายให้กับระบบ

4.2.1.3บันทึกผลการทดลอง

4.2.1.4นำผลการทดลองที่ได้ไป plot กราฟ แสดงดังรูปที่ 4.2



รูปที่ 4.2 แสดงผลการควบคุมอุณหภูมิ ที่ 70 องศาเซลเซียสโดยป้อนพิกัดแรงดันที่ แตกต่างกัน

4.2.2 สรุปผลการทดลอง

เมื่อมีการจ่ายแรงดันที่พิกัดต่างกันและทำการควบคุมอุณหภูมิเป้าหมายที่อุณหภูมิเดียวกัน แล้วผลที่ได้แสดงดังกราฟรูปที่ 4.2 ซึ่งจากกราฟจะเห็นได้ว่าที่พิกัดแรงดัน 100 % จะทำการควบคุมอุณหภูมิได้ยากมากเนื่องจากเมื่อมีการป้อนแรงดันสูงพลังงานความร้อนที่เกิดขึ้นทำให้การ

เพิ่มอุณหภูมิเป็นไปอย่างรวดเร็วจนไม่สามารถที่จะควบคุมที่เกินเป้าหมายได้ทันจึงทำให้เกิดค่า over shoot ที่สูงมาก และเมื่อระบบทำการควบคุมอุณหภูมิให้เข้าใกล้อุณหภูมิเป้าหมายแล้วค่าของอุณหภูมิจะแกว่งที่ซึ่งระบบทำการควบคุมได้ยากมากแต่เมื่อเปรียบเทียบกับ การป้อนค่าพิกัด แรงดันที่ 75% ถึงแม้จะมีค่า rise time ที่มากกว่าแต่มีค่า Over shoot ที่น้อยกว่าทำให้ระบบสามารถควบคุมอุณหภูมิได้ดีกว่ามากส่วนการป้อนค่าพิกัดแรงดันที่ 50 % ถือว่าไม่เป็นประโยชน์ต่อการควบคุมของระบบเนื่องจากการเพิ่มอุณหภูมิใช้เวลานาน(rise time สูง) และมีการแกว่งของอุณหภูมิ ขณะเข้าอุณหภูมิเป้าหมายช้ามากซึ่งถือว่าให้ผลตอบสนองต่อระบบช้า ระบบไม่สามารถควบคุมการเพิ่มอุณหภูมิได้ ดังนั้นสรุปได้ว่าการป้อนพิกัดแรงดันที่ 75 % จะทำให้สามารถควบคุมการเพิ่มอุณหภูมิได้ดีที่สุด

4.3 การทดลองที่ 3 การควบคุมการเพิ่มอุณหภูมิที่อุณหภูมิเป้าหมายต่างๆ

ในการควบคุมอุณหภูมิ ระบบต้องสามารถควบคุมอุณหภูมิเป้าหมายได้ทุกค่าในช่วงของการควบคุมซึ่งในการทดลองนี้ได้มีการกำหนดค่าอุณหภูมิเป้าหมาย ไว้ 3 ค่า คือ

- 1.อุณหภูมิเป้าหมายที่ 70 องศาเซลเซียส
- 2.อุณหภูมิเป้าหมายที่ 60 องศาเซลเซียส
- 3.อุณหภูมิเป้าหมายที่ 50 องศาเซลเซียส

ซึ่งทั้ง 3 กรณีจะเป็นการควบคุมการเพิ่มอุณหภูมิจากอุณหภูมิห้องคือประมาณ 30 องศาเซลเซียส ไปจนถึงอุณหภูมิเป้าหมาย

4.3.1 ขั้นตอนการทดลอง

- 4.3.1.1จ่ายไฟฟ้ากระแสสลับให้กับเตาอบ และจ่ายไฟฟ้ากระแสตรงให้กับระบบ
- 4.3.1.2กำหนดค่าอุณหภูมิเริ่มต้นให้กับระบบ
- 4.3.1.3ป้อนอุณหภูมิเป้าหมายให้กับระบบ
- 4.3.1.4บันทึกผลการทดลอง
- 4.3.1.5นำผลการทดลองที่ได้ไป plot กราฟ แสดงดังรูปที่ 4.3

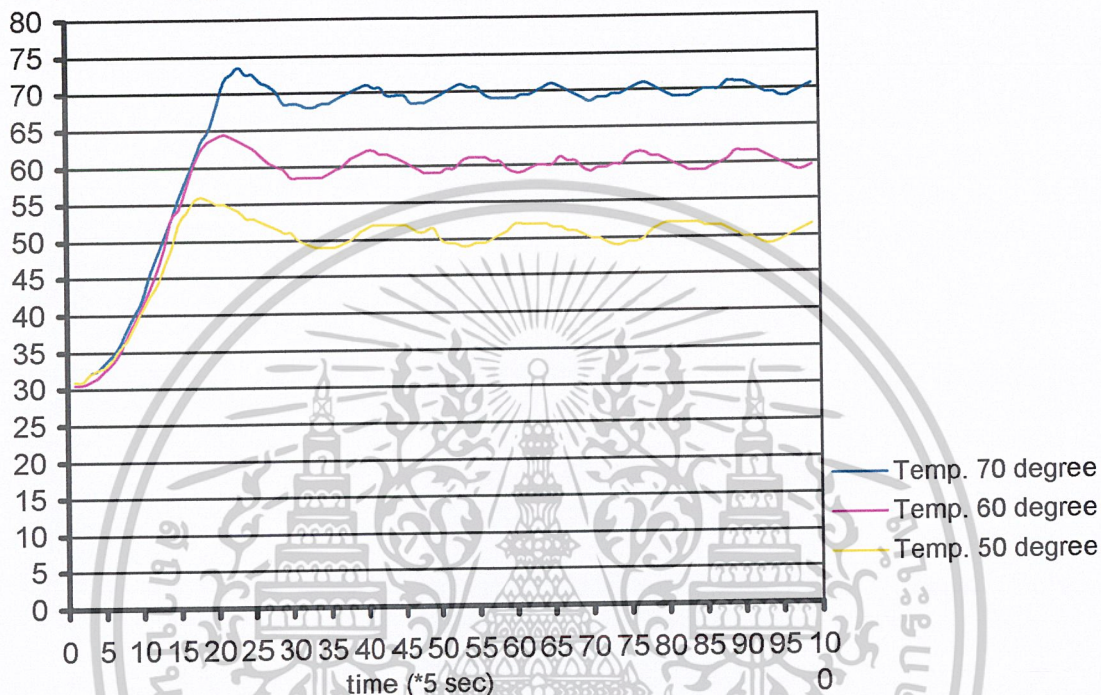
4.3.2 สรุปผลการทดลอง

จากรูปที่ 4.3 แสดงให้เห็นว่าถึงแม้มีการกำหนดค่าอุณหภูมิเป้าหมายที่แตกต่างกัน แต่เมื่อถูกควบคุมภายใต้ระบบควบคุมเดียวกัน จะให้กราฟที่มีคุณลักษณะคล้ายกันคือ มี ค่า rise time , Over shootใกล้เคียงกันมีการแกว่งของกราฟซึ่งแสดงถึงความสามารถในการตอบสนองของระบบที่ใกล้เคียงกันดังรูป แต่เนื่องจาก ระบบนี้มีควบคุมระบบแบบเปิดปิด (ON-OFF CONTROL)

ดังนั้นจึงมีค่าผิดพลาดอยู่บ้างซึ่งจากผลการทดลองจะมีค่าผิดพลาดอยู่ในช่วงไม่เกิน 2 องศา

เซลเซียส

Temperature. (degree celcius)



รูปที่ 4.3 แสดงผลการควบคุมอุณหภูมิที่อุณหภูมิเป้าหมายต่างๆ

4.4 การทดลองที่ 4 การควบคุมการลดอุณหภูมิที่อุณหภูมิเป้าหมายต่างๆ

ในการควบคุมการลดอุณหภูมินั้นแตกต่างจากการเพิ่มอุณหภูมิเนื่องจากใช้พัดลมระบายอากาศ(Exhaust Fan)และพัดลมดูดอากาศ(Inhaust Fan)เพื่อระบายอากาศที่มีอุณหภูมิสูงออกจากระบบและดูดอากาศที่มีอุณหภูมิต่ำเข้ามาในระบบทำให้อุณหภูมิของระบบลดลงและควบคุมให้เข้าสู่อุณหภูมิเป้าหมายได้อย่างดีที่สุด ซึ่งในการทดลองนี้ได้มีการกำหนดค่าอุณหภูมิเป้าหมายไว้ 3 ค่า คือ

- 1.อุณหภูมิเป้าหมายที่ 30 องศาเซลเซียส
- 2.อุณหภูมิเป้าหมายที่ 40 องศาเซลเซียส
- 3.อุณหภูมิเป้าหมายที่ 50 องศาเซลเซียส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งทั้ง 3 กรณีจะเป็นการควบคุมการลดอุณหภูมิจากอุณหภูมิที่สูงสุดคือประมาณ 70 องศาเซลเซียสไปจนถึงอุณหภูมิเป้าหมาย

4.4.1 ขั้นตอนการทดลอง

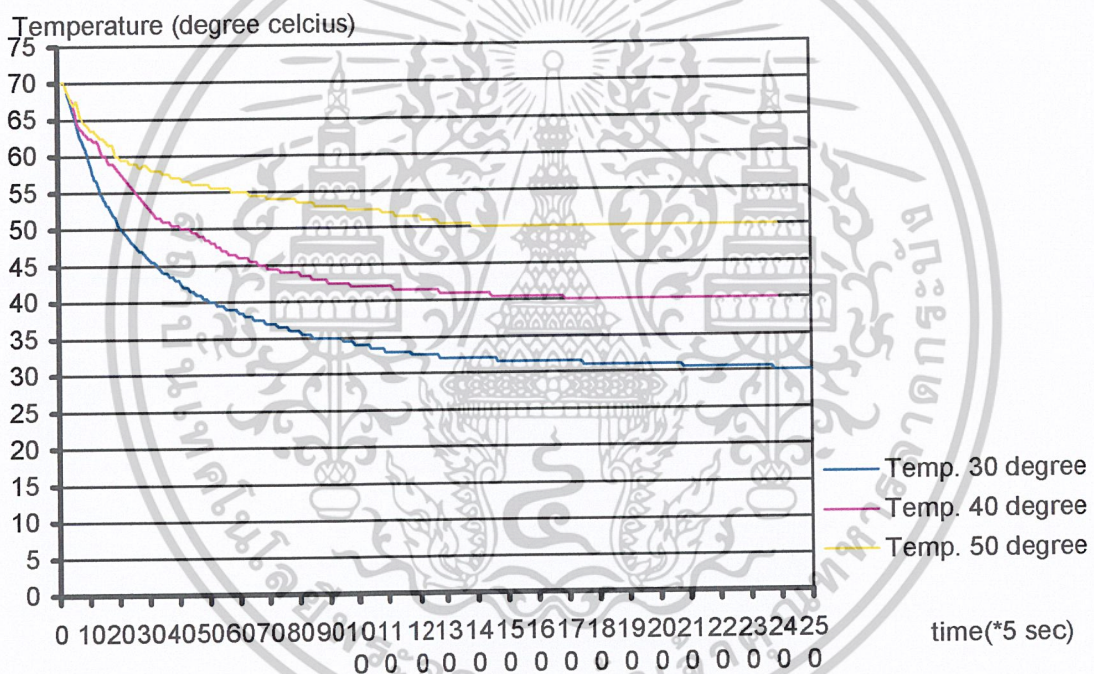
4.4.1.1 จ่ายไฟฟ้ากระแสสลับให้กับเตาอบ และจ่ายไฟฟ้ากระแสตรงให้กับระบบ

4.4.1.2 กำหนดค่าอุณหภูมิเริ่มต้นให้กับระบบ

4.4.1.3 ป้อนอุณหภูมิเป้าหมายให้กับระบบ

4.4.1.4 บันทึกผลการทดลอง

4.4.1.5 นำผลการทดลองที่ได้ไป plot กราฟ แสดงดังรูปที่ 4.4



รูปที่ 4.4 แสดงผลการทดลองการควบคุมการลดอุณหภูมิที่อุณหภูมิเป้าหมายต่างๆ

4.4.2 สรุปผลการทดลอง

การควบคุมการลดอุณหภูมิปัจจัยที่สำคัญคือ พัดระบายอากาศ(Exhaust Fan)และพัดลมดูดอากาศ(Inhaust Fan) ซึ่งจะต้องทำงานให้สัมพันธ์กัน แต่เมื่อนำมาควบคุมการลดอุณหภูมิ จะใช้เวลาในการควบคุมอุณหภูมินานมาก แสดงดังรูปที่4.4 ทั้งนี้ขึ้นอยู่กับความสามารถในการดูดอากาศและระบายอากาศ ของพัดลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 การทดลองที่ 5 การหาประสิทธิภาพของตัวทำความชื้น ชนิด อัลตราโซนิก

การทำความชื้นโดยใช้ตัวทำความชื้นชนิด อัลตราโซนิก (Ultrasonics Humidity) นั้นจะทำได้โดยนำตัวทำความชื้นมาจุ่มลงในน้ำ และจ่ายไฟให้กับตัวทำความชื้น จากนั้นตัวทำความชื้นก็จะทำความชื้นให้กับระบบด้วยหลักการดังที่ได้อธิบาย ในบทที่ 2

4.5.1 ขั้นตอนการทดลอง

4.5.1.1 จ่ายไฟฟ้ากระแสตรงให้กับตัวทำความชื้น

4.5.1.2 กำหนดความชื้นเริ่มต้นให้กับระบบ โดยการให้ความร้อนกับระบบเพื่อลดความชื้น

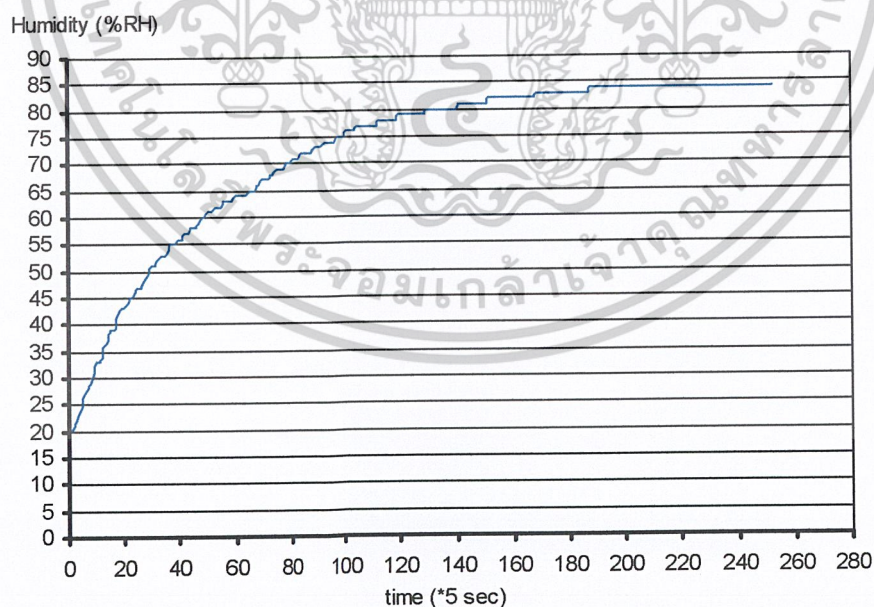
4.5.1.3 ป้อนความชื้นเป้าหมายให้กับระบบ

4.5.1.4 บันทึกผลการทดลอง

4.5.1.5 นำผลการทดลองที่ได้ไป plot กราฟ แสดงดังรูปที่ 4.5

4.5.2 สรุปผลการทดลอง

ในการหาประสิทธิภาพของตัวทำความชื้นชนิด อัลตราโซนิก นั้น ขึ้นอยู่กับระดับน้ำ หากระดับน้ำที่ตัวทำความชื้นจุ่มอยู่มีมากเกินไปจะทำให้ตัวทำความชื้นทำความชื้นได้น้อย และเมื่อทำความชื้นในเวลานานจะทำให้ระดับน้ำลดลงต่ำกว่าระดับตัวทำความชื้นจะทำให้ไม่สามารถสร้างความชื้นได้ และอาจทำให้ตัวทำความชื้นเสียหายได้ จากรูปที่ 4.5 แสดงให้เห็นว่าเมื่อเวลาผ่านไปจะทำให้อัตราการเพิ่มความชื้นลดลง



รูปที่ 4.5 แสดงผลการหาประสิทธิภาพของตัวทำความชื้นชนิด อัลตราโซนิก

4.6 การทดลองที่ 6 การควบคุมความชื้นสัมพัทธ์ที่ความชื้นเป้าหมายต่างๆ

การควบคุมความชื้นสัมพัทธ์ขึ้นอยู่กับปัจจัยหลายอย่างเช่น ประสิทธิภาพของตัวทำความชื้น ประสิทธิภาพในการวัดสัญญาณตัววัดสัญญาณความชื้น การกำหนดค่าความชื้นเริ่มต้นของระบบ ประสิทธิภาพของพัดลมดูดอากาศซึ่งจะเป็นตัวดูดความชื้นจากตัวทำความชื้นเข้ามาสู่ระบบ ประสิทธิภาพของตัวทำความร้อน (heater) ซึ่งมีหน้าที่ในการลดความชื้น ในกรณีที่ความชื้นในระบบมีค่าเกินความชื้นเป้าหมายที่กำหนดไว้ ซึ่งในการทดลองนี้ได้มีการกำหนดความชื้นสัมพัทธ์เป้าหมายไว้ 2 ค่า คือ

1.ความชื้นสัมพัทธ์ที่ 50 %RH

2.ความชื้นสัมพัทธ์ที่ 70 %RH

แต่เนื่องจากตัววัดสัญญาณความชื้นมีค่าความไวสูงดังนั้นจึงต้องมีการทดลองซ้ำ หลายครั้ง เพื่อให้ได้ผลการทดลองที่ใกล้เคียงกันมากที่สุด

4.6.1 ขั้นตอนการทดลอง

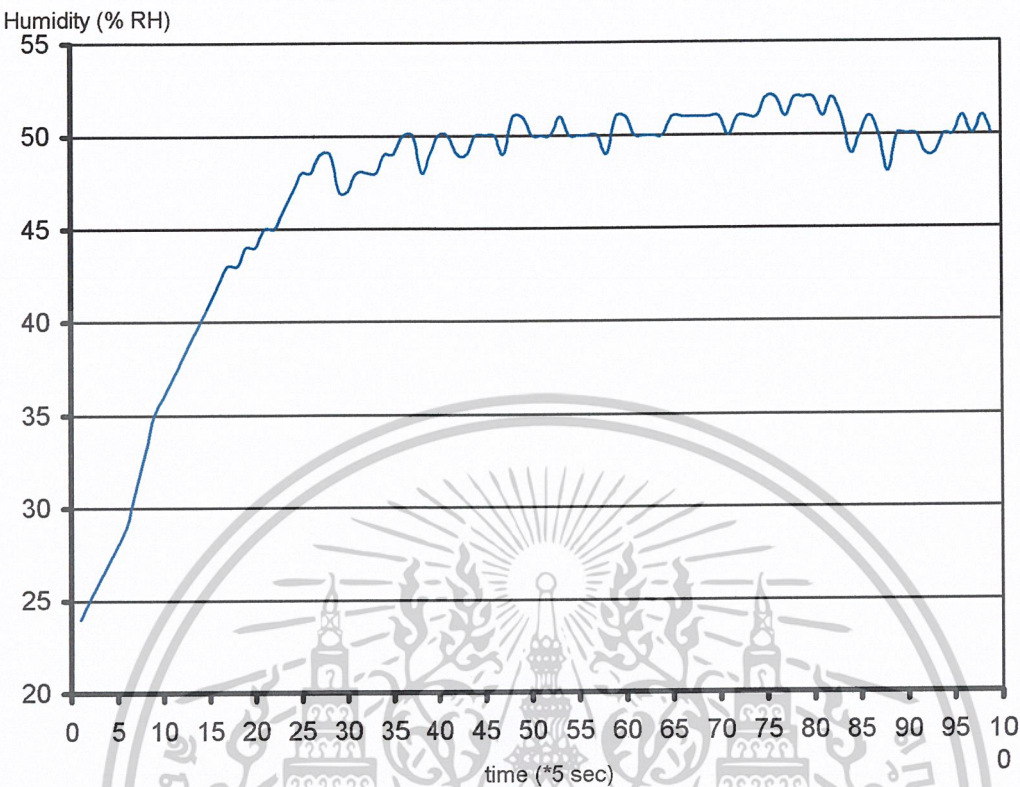
4.6.1.1.จ่ายไฟฟ้ากระแสสลับให้กับเตาอบ และจ่ายไฟฟ้ากระแสตรงให้กับระบบ

4.6.1.2กำหนดความชื้นเริ่มต้นให้กับระบบ โดยการให้ความร้อนกับระบบเพื่อลดความชื้น

4.6.1.3ป้อนค่าความชื้นสัมพัทธ์ที่ต้องการให้กับระบบ

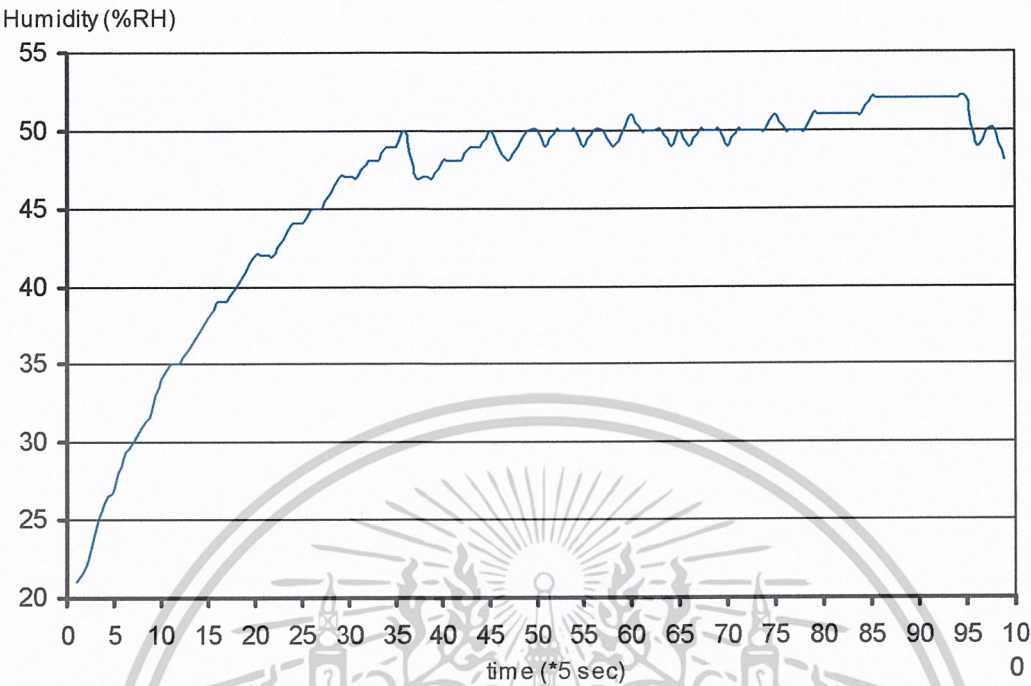
4.6.1.4บันทึกผลการทดลอง

4.6.1.5นำผลการทดลองที่ได้ไป plot กราฟ แสดงดังรูปที่ 4.6,4.7,4.8,4.9

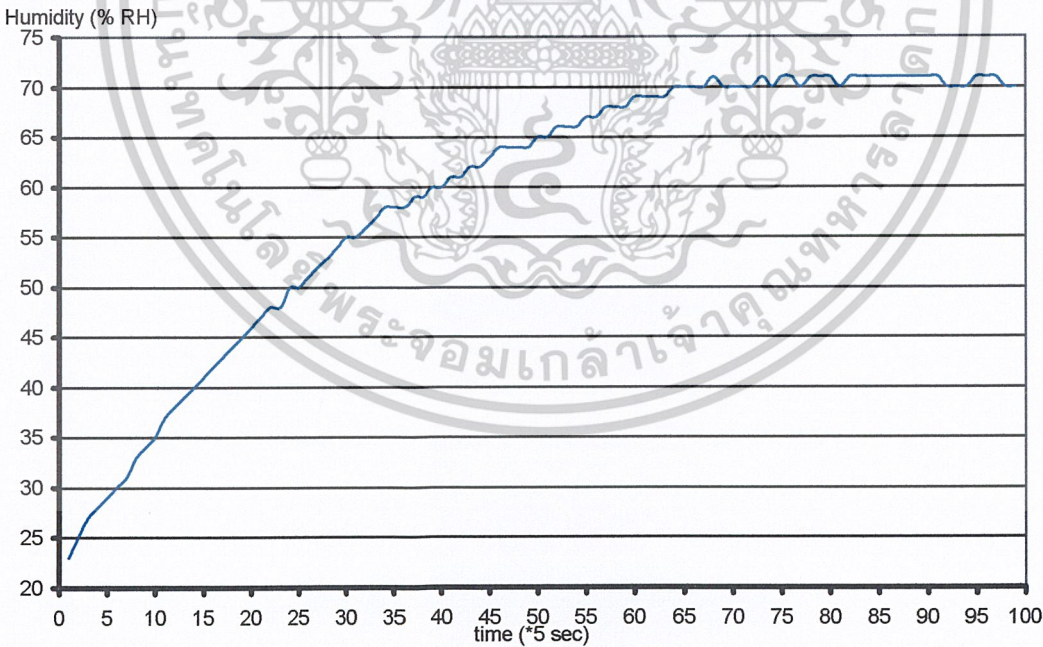


รูปที่ 4.6 แสดงผลการทดลองการควบคุมความชื้นที่ความชื้นเป้าหมาย 50 % RH (การทดลองครั้งที่ 1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

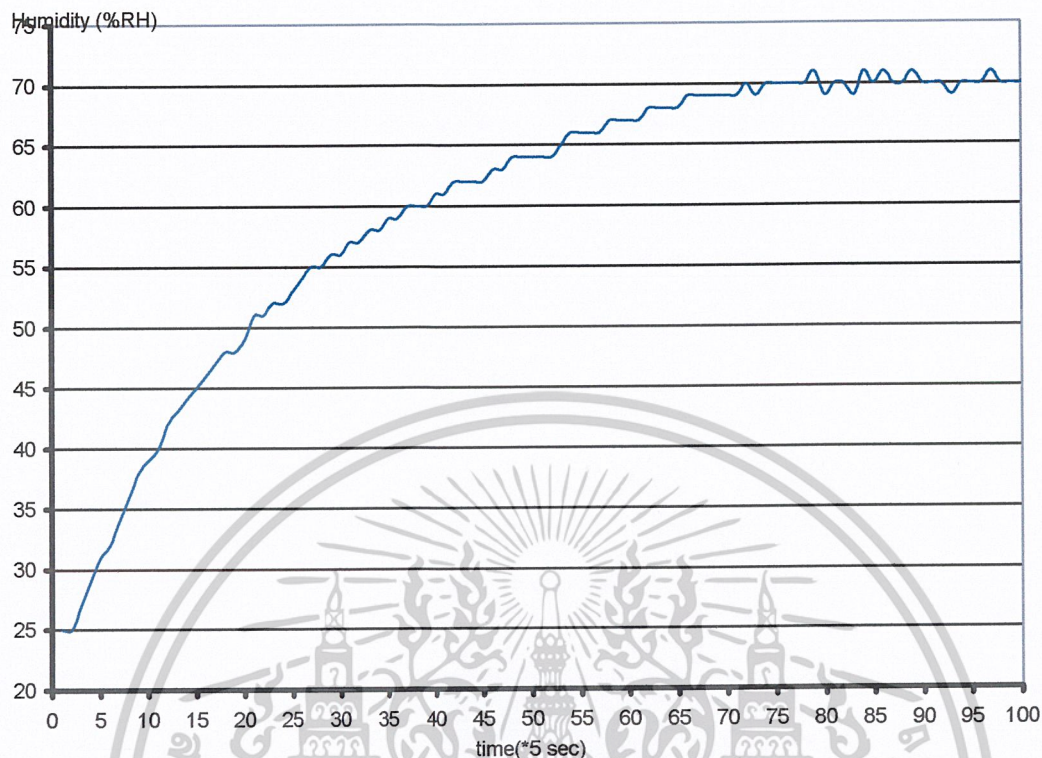


รูปที่ 4.7 แสดงผลการทดลองการควบคุมความชื้นที่ความชื้นเป้าหมาย 50 % RH
(การทดลองครั้งที่ 2)



รูปที่ 4.8 แสดงผลการทดลองการควบคุมความชื้นที่ความชื้นเป้าหมาย 70 % RH
(การทดลองครั้งที่ 1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 แสดงผลการทดลองการควบคุมความชื้นที่ความชื้นเป้าหมาย 70 % RH (การทดลองครั้งที่ 2)

4.6.2 สรุปผลการทดลอง

จากผลการทดลอง การควบคุมความชื้นสัมพัทธ์ให้ได้ตามความชื้นเป้าหมายนั้น ระบบสามารถควบคุมได้แต่ไม่ดีมากนัก เนื่องจากเกิดค่าผิดพลาดซึ่งเห็นได้จากการแกว่งของกราฟ โดยมีผลมาจากการปรับระดับความเร็วของพัดลมระบายอากาศ และการควบคุมการเปิดปิดตัวทำความร้อน ซึ่งมีเมื่อผ่านการวัดสัญญาณ โดยตัววัดความชื้นที่มีความไวสูงแล้วผลการทดลองจึงแสดงได้ดังรูปที่ 4.6, 4.7, 4.8, 4.9

4.7 การทดลองที่ 7 การควบคุมอุณหภูมิและความชื้นสัมพัทธ์ให้ได้ผลตามต้องการ

ในการควบคุมอุณหภูมิและความชื้นสัมพัทธ์นั้นขึ้นอยู่กับ การสร้างความสัมพันธ์ระหว่างปัจจัยต่างๆ ที่มีผลต่อการควบคุมอุณหภูมิและ ความชื้นและให้มีความสัมพันธ์กับตัววัดสัญญาณด้วย จากนั้นนำความสัมพันธ์ต่างๆ ไปเขียนเป็นโปรแกรมควบคุมอัตโนมัติซึ่งในการทดลองนี้ได้แบ่งการทดลองเป็น 3 กรณี คือ

1. กรณีป้อนค่าอุณหภูมิเป้าหมาย และความชื้นสัมพัทธ์ เป้าหมาย ให้มีค่าเท่ากันซึ่งทำการพิจารณาที่ 2 ค่าเป้าหมายคือ

- อุณหภูมิเป้าหมายที่ 50 องศาเซลเซียสและความชื้นสัมพัทธ์เป้าหมายที่ 50 %

RH

- อุณหภูมิเป้าหมายที่ 65 องศาเซลเซียสและความชื้นสัมพัทธ์เป้าหมายที่ 65 %

RH

2. กรณีป้อนค่าอุณหภูมิเป้าหมายมีค่ามากกว่า ความชื้นสัมพัทธ์ เป้าหมายซึ่งทำการป้อนอุณหภูมิ เป้าหมายที่ 60 องศาเซลเซียสและความชื้นสัมพัทธ์เป้าหมายที่ 15 % RH

3. กรณีป้อนค่าอุณหภูมิเป้าหมายมีค่าน้อยกว่า ความชื้นสัมพัทธ์ เป้าหมายซึ่งทำการป้อนอุณหภูมิ เป้าหมายที่ 35 องศาเซลเซียสและความชื้นสัมพัทธ์เป้าหมายที่ 50 % RH

4.7.1. ขั้นตอนการทดลอง

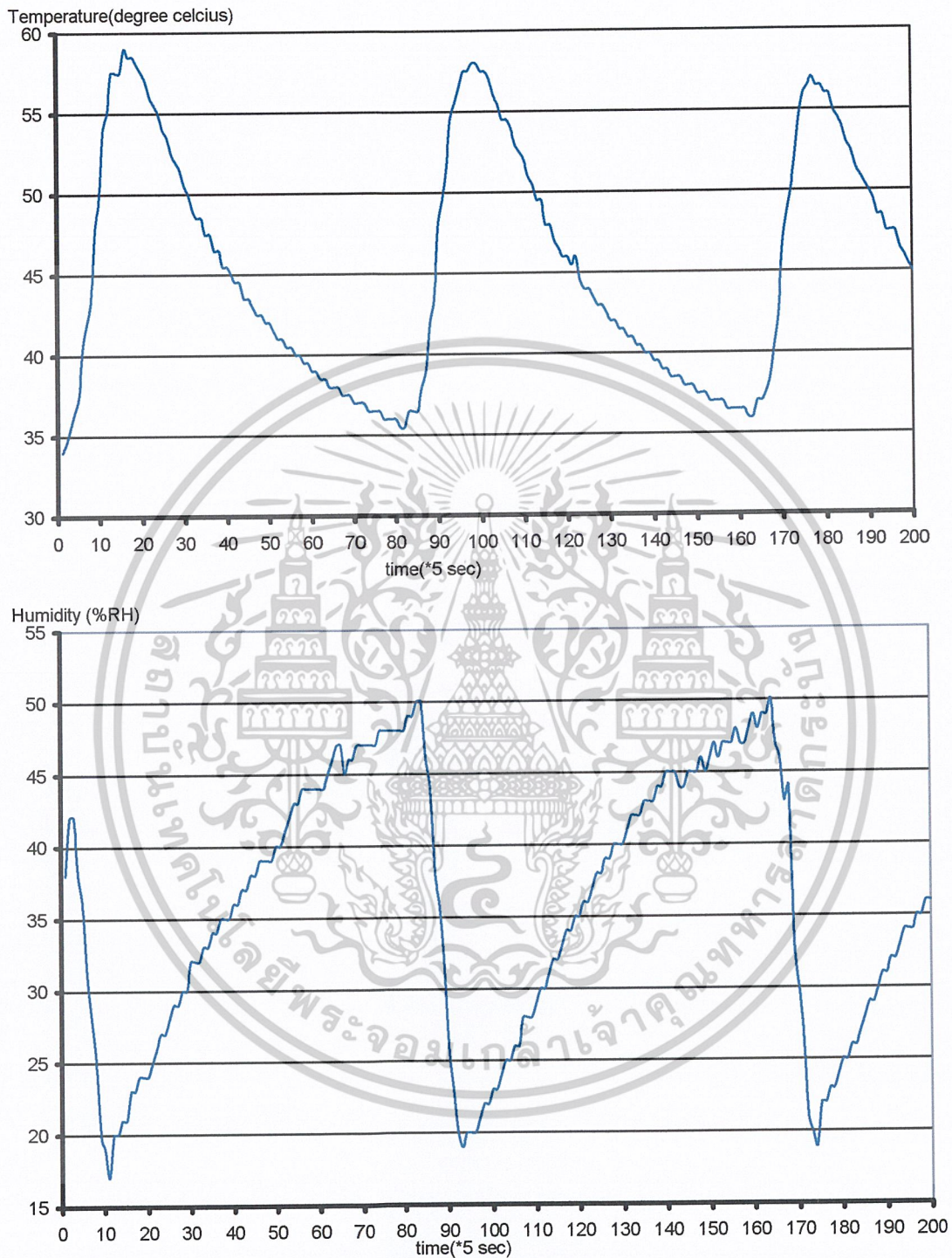
4.7.1.1 จ่ายไฟฟ้ากระแสสลับให้กับเตาอบ และจ่ายไฟฟ้ากระแสตรงให้กับระบบ

4.7.1.2 กำหนดค่าอุณหภูมิและความชื้นเริ่มต้นให้กับระบบ

4.7.1.3 ป้อนอุณหภูมิและความชื้นสัมพัทธ์เป้าหมายให้กับระบบ

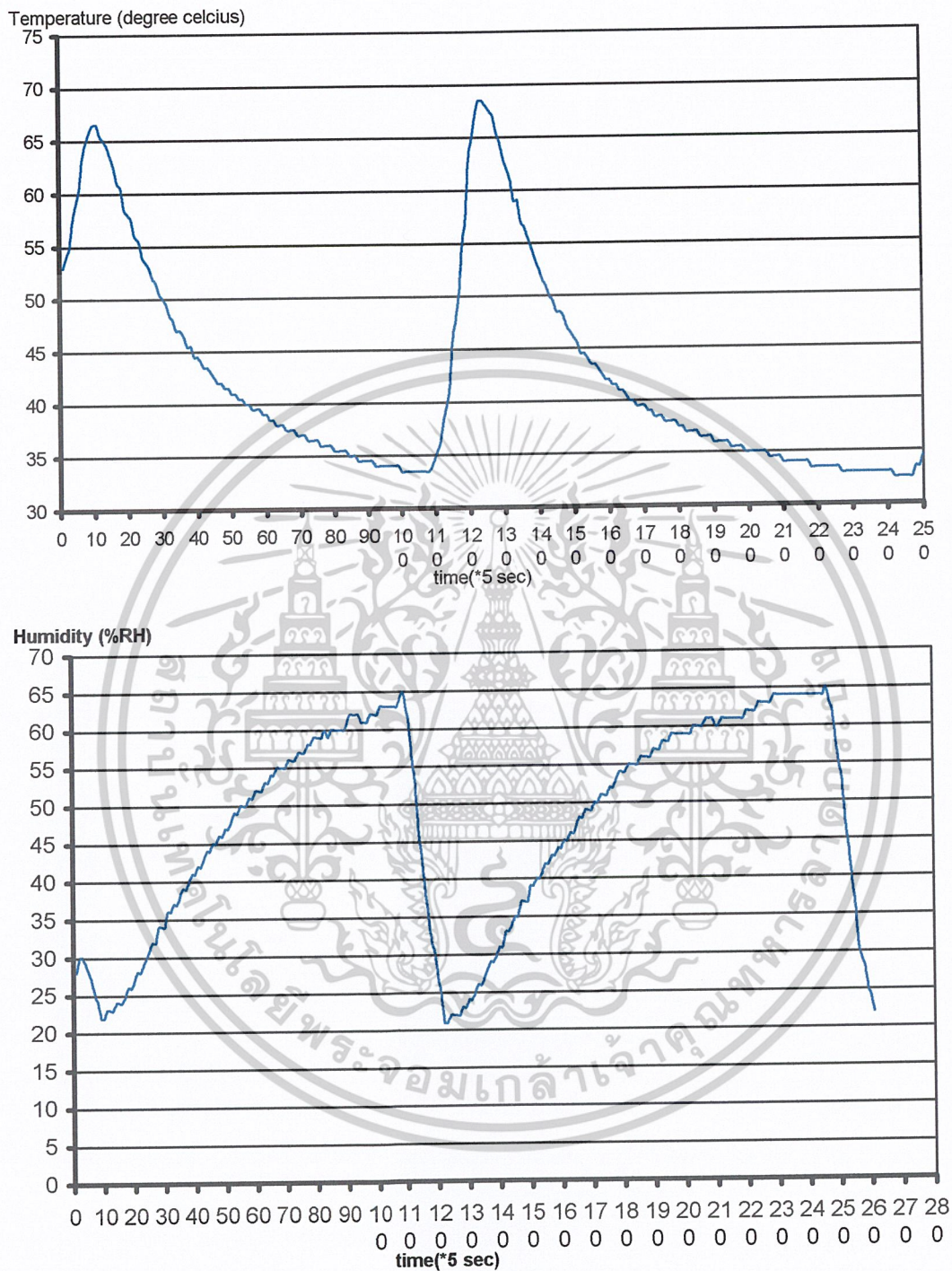
4.7.1.4 บันทึกผลการทดลอง

4.7.1.5 นำผลการทดลองที่ได้ไป plot กราฟ แสดงดังรูปที่ 4.10, 4.11, 4.12, 4.13



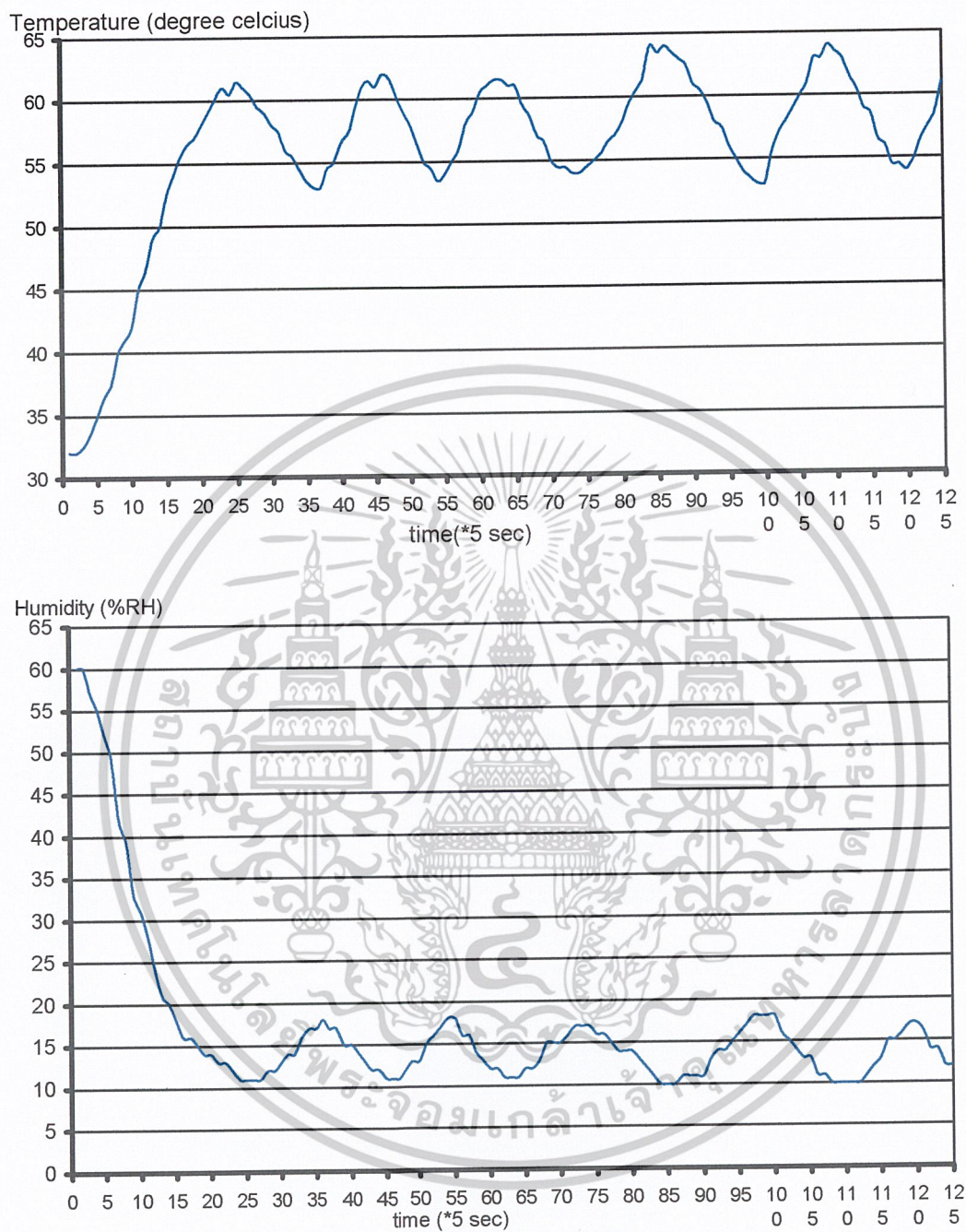
รูปที่ 4.10 แสดงผลการทดลองการควบคุมอุณหภูมิและความชื้นสัมพัทธ์ที่มีค่าเท่ากัน
กรณีป้อนค่าอุณหภูมิเป้าหมายที่ 50 องศาเซลเซียสและความชื้นสัมพัทธ์เป้าหมายที่ 50 % RH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



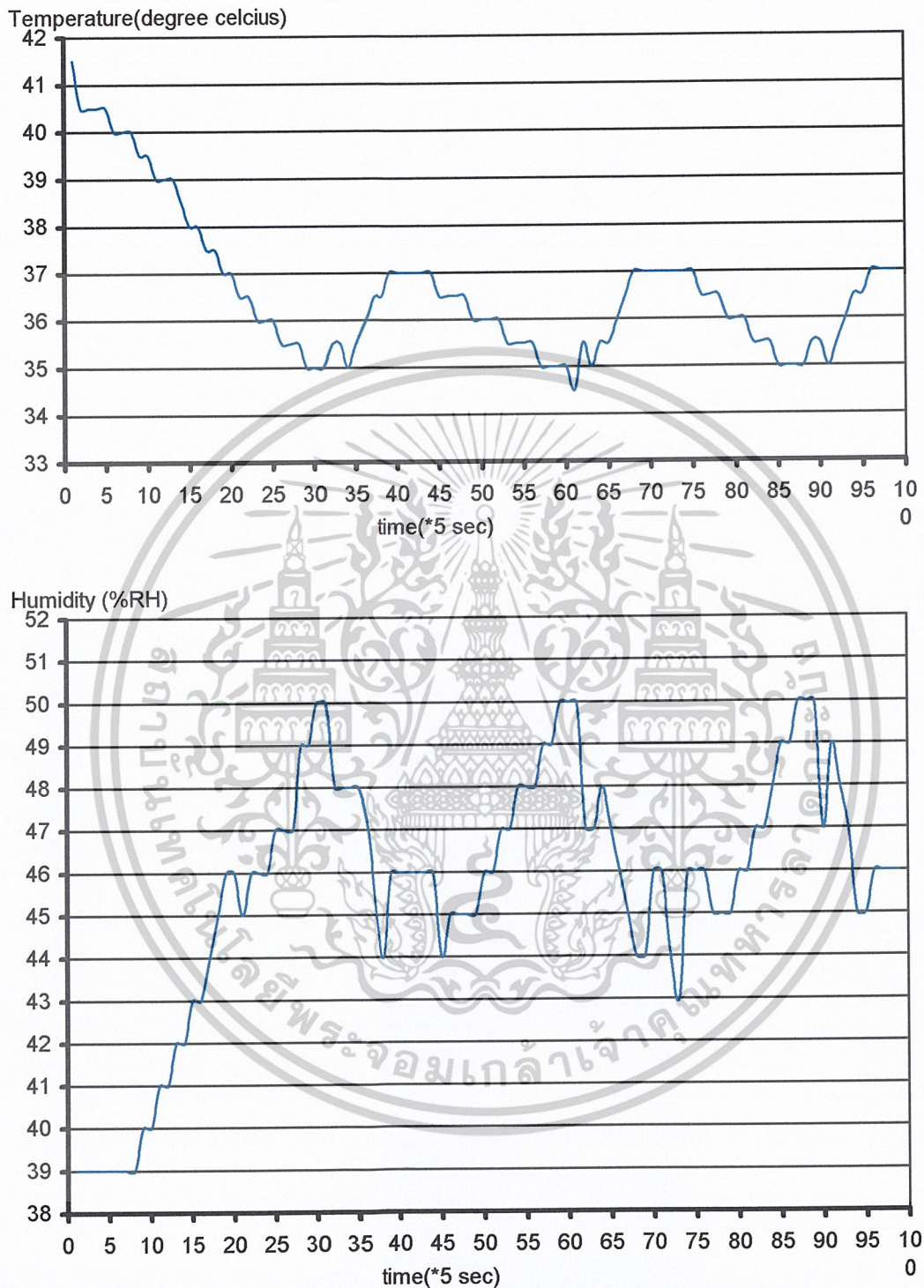
รูปที่ 4.11 แสดงผลการทดลองการควบคุมอุณหภูมิและความชื้นสัมพัทธ์ที่มีค่าเท่ากัน
กรณีเปลี่ยนค่าอุณหภูมิเป้าหมายที่ 65 องศาเซลเซียสและความชื้นสัมพัทธ์เป้าหมายที่ 65 % RH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 แสดงผลการทดลองการควบคุมอุณหภูมิที่มีค่ามากกว่าความชื้นสัมพัทธ์
โดยป้อนค่าอุณหภูมิเป้าหมายที่ 60 องศาเซลเซียสและความชื้นสัมพัทธ์เป้าหมายที่ 15 % RH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 แสดงผลการทดลองการควบคุมอุณหภูมิที่มีค่าน้อยกว่าความชื้นสัมพัทธ์
โดยป้อนค่าอุณหภูมิเป้าหมายที่ 35 องศาเซลเซียสและความชื้นสัมพัทธ์เป้าหมายที่ 50 % RH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7.2 สรุปผลการทดลอง

จากผลการทดลอง ควบคุมอุณหภูมิและความชื้นสัมพัทธ์ในกรณีต่างๆแล้ว แสดงให้เห็นว่าการเปลี่ยนแปลงของอุณหภูมิและความชื้นมีผลต่อกันและกันตลอดเวลาโดยจากรูปที่ 4.10 และ 4.11 แสดงให้เห็นอย่างชัดเจนว่า การเพิ่มอุณหภูมิจะทำให้ความชื้นสัมพัทธ์ลดลงและการเพิ่มความชื้นสัมพัทธ์ก็จะทำให้อุณหภูมิลดลงเช่นเดียวกันซึ่งโดยสภาพแวดล้อมปกติแล้วเมื่อมีอุณหภูมิสูง ความชื้นสัมพัทธ์จะต่ำ จึงเป็นไปได้ที่เมื่ออุณหภูมิสูงแล้วความชื้นจะสูงหรือมีค่าเท่ากัน ดังนั้นในกรณีนี้ไม่สามารถที่จะควบคุมสภาพแวดล้อมได้

ในกรณีที่มีการควบคุมสภาพแวดล้อมที่มีอุณหภูมิสูง ความชื้นสัมพัทธ์ต่ำ ซึ่งผลการทดลองในรูปที่ 4.12 แสดงให้เห็นว่าเมื่อเริ่มต้นการทำงาน ระบบจะควบคุมการเพิ่มอุณหภูมิให้ถึงเป้าหมายก่อนและในขณะเดียวกันความชื้นสัมพัทธ์ก็จะลดลงจากนั้นระบบจะควบคุมการเพิ่มความชื้นให้ถึงจุดมุ่งหมาย ซึ่งจะทำให้อุณหภูมิลดลงระบบจึงต้องทำการเพิ่มอุณหภูมิให้ถึงเป้าหมายอีกครั้งโดยระบบจะทำการควบคุมเช่นนี้ซ้ำไปจนกระทั่งค่าอุณหภูมิและความชื้นสัมพัทธ์เข้าใกล้เป้าหมายที่สุด

ส่วนในกรณีที่มีการควบคุมสภาพแวดล้อมที่มีอุณหภูมิต่ำ ความชื้นสัมพัทธ์สูง ก็สามารถอธิบายผลการทดลองได้เช่นเดียวกัน คือในช่วงแรกของการควบคุมระบบจะทำการลดอุณหภูมิลงจนถึงอุณหภูมิเป้าหมาย ซึ่งจะทำให้มีความชื้นเพิ่มขึ้น และเมื่ออุณหภูมิลดลงจนเกินค่าเป้าหมายระบบจึงต้องมีการเพิ่มอุณหภูมิขยับให้ได้ค่าใกล้เคียงเป้าหมายมากที่สุดจึงทำให้ความชื้นลดลงจากนั้นระบบจะทำการเพิ่มความชื้นสัมพัทธ์ให้ได้ค่าเป้าหมายที่ต้องการ โดยที่ระบบจะควบคุมซ้ำไปมาเช่นนี้เรื่อยไปจนกระทั่งได้อุณหภูมิและความชื้นสัมพัทธ์ใกล้เคียงเป้าหมาย ดังผลการทดลองในรูปที่ 4.13

บทที่ 5

บทวิจารณ์และสรุป

5.1 ปัญหาที่เกิดขึ้น

5.1.1 ปัญหาการหน่วงเวลา(Time constant) ของตัวทำความร้อน (heater) ทำให้ไม่สามารถกำหนดระยะเวลาช่วงของสัญญาณ พัลส์ที่จะส่งไปยังตัวทำความร้อนได้ จึงทำให้เกิดค่าผิดพลาดในการควบคุมอุณหภูมิ

5.1.2 คุณลักษณะของพัดลมดูดอากาศ(Inhaust Fan) และพัดลมระบายอากาศ(Exhaust Fan) มีผลอย่างมากต่อการลดอุณหภูมิและการเพิ่มความชื้น ซึ่งการลดอุณหภูมิและการเพิ่มความชื้นของระบบ เป็นไปได้อย่างช้ามากเนื่องจากการดูดอากาศ และการระบาย อากาศของพัดลมเป็นไปได้อย่างช้า

5.1.3 ตัวแปรที่ถูกควบคุมแบบ ON/OFF จะมีการแกว่งที่ Steady State

5.1.4 มีสัญญาณรบกวนในระบบควบคุมอัตโนมัติ

5.1.5 sensor ที่ได้นำมาใช้ใน project นี้ (DS1820, SHT1x) จะใช้ระยะเวลาค่าหนึ่งในการคำนวณและ ส่งค่า ทำให้ค่าที่วัดและอ่านได้ช้ากว่าค่าที่แท้จริง

5.1.6 ค่าที่ได้ออกมายังไม่ใช่ค่าที่แท้จริง เนื่องจากใช้ sensor วัดแค่ตัวเดียว จึงวัดค่าได้ไม่ทั่วทั้งระบบ

5.2 แนวทางการพัฒนา

5.2.1 ทำการทดลองในเงื่อนไขเดิมซ้ำกันหลายๆครั้งเพื่อหาแนวทางในการควบคุมที่เหมาะสมกับเงื่อนไขนั้นๆ

5.2.2 เพิ่มตัวsensorให้มากขึ้นเพื่อให้ค่าที่ออกมาใกล้เคียงกับค่าที่แท้จริงในระบบ

5.2.3 เปลี่ยนตำแหน่งของพัดลมดูดอากาศและพัดลมระบายอากาศและตัวสร้างความชื้นให้มีผลต่อระบบให้มากที่สุด

5.2.4 เขียนโปรแกรมให้มีการรูดคุมเพื่อให้สามารถรองรับได้ทุกเงื่อนไขที่จะเกิดขึ้น

5.3 สรุปและวิจารณ์

การควบคุมอุณหภูมิและความชื้นสัมพัทธ์ให้ได้ตามเป้าหมายนั้นขึ้นอยู่กับปัจจัยหลายอย่าง เช่น การควบคุมการเพิ่มอุณหภูมิของตัวทำความร้อน การควบคุมพัดลมระบายอากาศ

และพัดลมดูดอากาศ ความสามารถในการวัดสัญญาณของตัววัดสัญญาณบนอุณหภูมิและความชื้นสัมพัทธ์ สภาพภายนอกของระบบ ซึ่งปัจจัยต่างๆ ล้วนแล้วแต่มีผลที่จะทำให้เกิดค่าผิดพลาด ในการควบคุมอุณหภูมิและความชื้นสัมพัทธ์ให้เข้าสู่ค่าเป้าหมายได้ตลอดเวลาดังนั้น ก่อนที่จะมีการควบคุมอุณหภูมิและความชื้นสัมพัทธ์จึงต้องมีการศึกษาถึงผลกระทบของปัจจัยต่างๆ ที่จะมีผลกระทบต่อการควบคุมระบบ เช่นในการควบคุมอุณหภูมินั้นตัวทำความร้อนมีความสำคัญมากต่อการควบคุมดังนั้นจึงต้องพยายามศึกษาผลกระทบที่เกิดจากปัจจัยต่างๆ แล้วทำการหาแนวทางในการแก้ไข เช่น ช่วงระยะเวลาหน่วงในการเปิดตัวทำความร้อนจนกระทั่งตัวทำความร้อนมีความร้อนเต็มที่จะมีผลอย่างมากต่อการควบคุมอุณหภูมิขณะเข้าใกล้ค่าอุณหภูมิเป้าหมาย หากมีการส่งสัญญาณเป็นพัลส์ก็จะต้องกำหนดระยะเวลาในช่วง 1 พัลส์ให้แน่นอนเพื่อจะทำให้ควบคุมอุณหภูมิ ให้เข้าสู่ค่าเป้าหมายได้อย่างง่ายขึ้น หรืออาจจะลดแรงดันไฟ เพื่อให้ตัวทำความร้อนให้พลังงานความร้อนลดลงซึ่งจะทำให้ควบคุมค่าการเปลี่ยนแปลงอุณหภูมิได้ง่ายขึ้น พัดลมดูดอากาศก็มีผลโดยตรงต่อการลดอุณหภูมิ ซึ่งอาจต้องใช้เวลานานมากในการลดอุณหภูมิ โดยใช้พัดลมดูดอากาศจากสภาพแวดล้อมภายนอกเข้ามาสู่ระบบ ส่วนตัววัดอุณหภูมิก็เป็นปัจจัยที่มีผลต่อการควบคุมอุณหภูมิเช่นเดียวกันเนื่องจากสัญญาณที่ได้จากการวัดจะถูกส่งไปยังหน่วยประมวลผลซึ่งช่วงเวลาในการอ่านค่าอุณหภูมิแต่ละครั้งต้องเหมาะสมเพื่อให้ได้ผลตอบสนองที่ดีในการนำค่าไปประมวลผล

ปัจจัยที่มีผลต่อการควบคุมความชื้นสัมพัทธ์ เช่นพัดลมดูดอากาศ ซึ่งจะควบคุมความชื้นที่เกิดจากตัวสร้างความชื้นเข้าสู่ระบบ ส่วนพัดลมระบายอากาศจะระบายความชื้นจากระบบสู่สภาพแวดล้อมภายนอก และตัวทำความร้อนมีผลกระทบโดยตรงต่อการลดความชื้นต้องให้พลังงานความร้อนที่พอดีเพื่อไม่ให้ความชื้นสัมพัทธ์ลดลงจนเกินเป้าหมายไปมาก ซึ่งจากการทดลองที่ผ่านมาจะแสดงให้เห็นการเพิ่มความชื้นใช้เวลานานแต่การลดความชื้นโดยใช้พลังงานความร้อนจากตัวทำความร้อนใช้เวลาน้อยมากซึ่งจะทำให้ไม่สามารถควบคุมความชื้นให้เข้าสู่เป้าหมายได้โดยง่ายและตัววัดความชื้นก็มีผลกระทบต่อการควบคุมความชื้นเช่นกัน หากตัววัดความชื้นที่มีประสิทธิภาพดี มีความแม่นยำสูง มีค่าความผิดพลาดน้อยก็จะทำให้ระบบสามารถควบคุมความชื้นให้เข้าสู่เป้าหมายได้อย่างดีขึ้น

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยความช่วยเหลือจากบุคคลหลายๆท่าน ทางผู้จัดทำ
ขอกราบขอบพระคุณ คุณพ่อ คุณแม่และญาติพี่น้องทุกท่านที่ให้การสนับสนุน และเป็นกำลังใจ
ให้แก่ผู้จัดทำตลอดมา

ขอกราบขอบพระคุณอาจารย์ที่ปรึกษา รศ.ดร.วันชัย ธีรวัชร และ นายรัชชัย คำศรี ที่
ได้กรุณาให้คำปรึกษา คำแนะนำพร้อมทั้งคณาจารย์ทุกท่านที่ได้ประสิทธิประสาทความรู้แก่
ผู้จัดทำ

ขอขอบพระคุณเพื่อนๆทุกคนที่ได้ให้คำแนะนำให้ข้มอุปกรณ์และ เป็นกำลังใจแก่ผู้จัดทำ

ผู้จัดทำ



เอกสารอ้างอิง

1. ชัยวัฒน์ ลีมพรจิตรวิไล, วรพจน์ กรแก้ววัฒนกุล, ”เรียนรู้และปฏิบัติการ ไมโครคอนโทรลเลอร์ MCS – 51 ”, อินโนเวทีฟ เอ็กเพอริเมนต์ , หน้า 273-274 , 283-290 , 399-405 :
2. วิทยา วัชรวิทยากุล ; “ ภาษาและการโปรแกรม C “, ซีเอ็ดยูเคชั่น ; 2521 , หน้า 16-140



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

28/40-Pin 8-Bit CMOS FLASH Microcontrollers

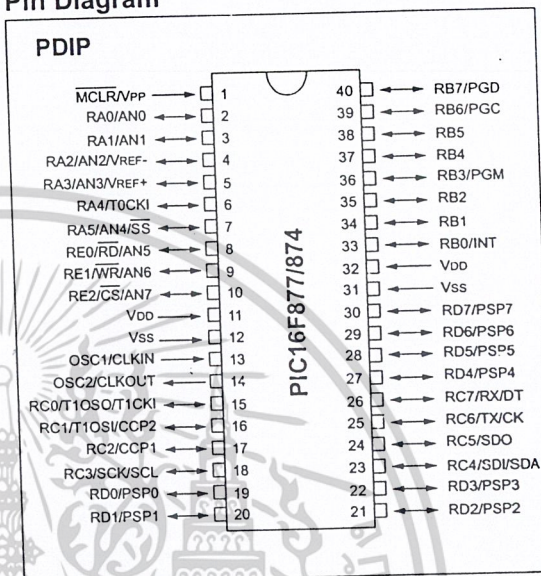
Devices Included in this Data Sheet:

- PIC16F873
- PIC16F876
- PIC16F874
- PIC16F877

Microcontroller Core Features:

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,
Up to 368 x 8 bytes of Data Memory (RAM)
Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and
Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC
oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM
technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP) via two
pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial, Industrial and Extended temperature
ranges
- Low-power consumption:
 - < 0.6 mA typical @ 3V, 4 MHz
 - 20 µA typical @ 3V, 32 kHz
 - < 1 µA typical standby current

Pin Diagram



Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,
can be incremented during SLEEP via external
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period
register, prescaler and postscaler
- Two Capture, Compare, PWM modules
 - Capture is 16-bit, max. resolution is 12.5 ns
 - Compare is 16-bit, max. resolution is 200 ns
 - PWM max. resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master
mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver
Transmitter (USART/SCI) with 9-bit address
detection
- Parallel Slave Port (PSP) 8-bits wide, with
external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for
Brown-out Reset (BOR)

PIC16F87X

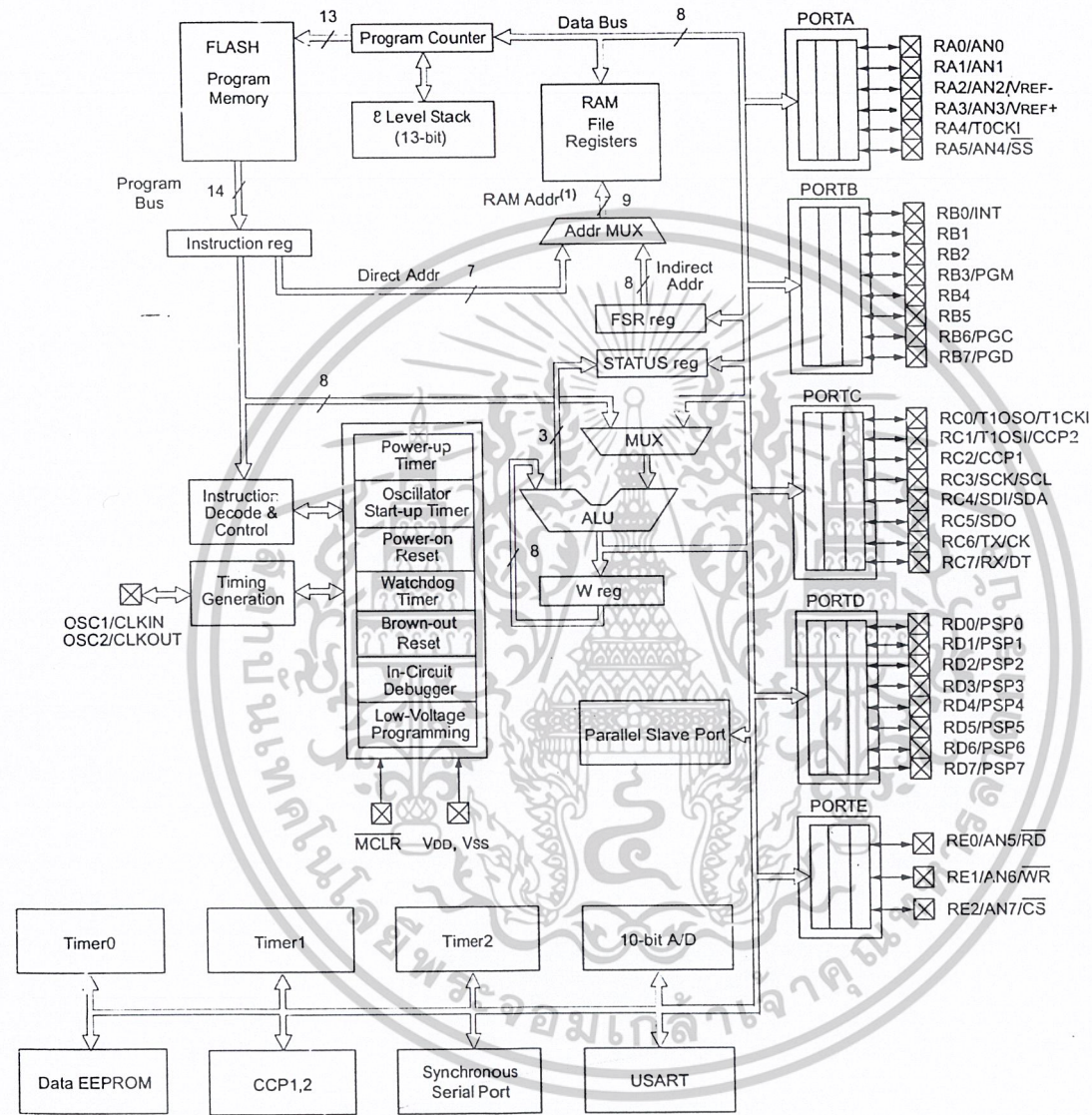
Key Features PICmicro™ Mid-Range Reference Manual (DS33023)	PIC16F873	PIC16F874	PIC16F876	PIC16F877
Operating Frequency	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz
RESETS (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
FLASH Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory	128	128	256	256
Interrupts	13	14	13	14
I/O Ports	Ports A,B,C	Ports A,B,C,D,E	Ports A,B,C	Ports A,B,C,D,E
Timers	3	3	3	3
Capture/Compare/PWM Modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Instruction Set	35 instructions	35 instructions	35 instructions	35 instructions

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIC16F87X

FIGURE 1-2: PIC16F874 AND PIC16F877 BLOCK DIAGRAM

Device	Program FLASH	Data Memory	Data EEPROM
PIC16F874	4K	192 Bytes	128 Bytes
PIC16F877	8K	368 Bytes	256 Bytes



Note 1: Higher order bits are from the STATUS register.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIC16F87X

TABLE 1-2: PIC16F874 AND PIC16F877 PINOUT DESCRIPTION

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS ⁽⁴⁾	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	O	--	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
MCLR/VPP	1	2	18	I/P	ST	Master Clear (Reset) input or programming voltage input. This pin is an active low RESET to the device.
RA0/AN0	2	3	19	I/O	TTL	PORTA is a bi-directional I/O port. RA0 can also be analog input0. RA1 can also be analog input1. RA2 can also be analog input2 or negative analog reference voltage. RA3 can also be analog input3 or positive analog reference voltage. RA4 can also be the clock input to the Timer0 timer/counter. Output is open drain type. RA5 can also be analog input4 or the slave select for the synchronous serial port.
RA1/AN1	3	4	20	I/O	TTL	
RA2/AN2/VREF-	4	5	21	I/O	TTL	
RA3/AN3/VREF+	5	6	22	I/O	TTL	
RA4/T0CKI	6	7	23	I/O	ST	
RA5/SS/AN4	7	8	24	I/O	TTL	
RB0/INT	33	36	8	I/O	TTL/ST ⁽¹⁾	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0 can also be the external interrupt pin. RB3 can also be the low voltage programming input. Interrupt-on-change pin. Interrupt-on-change pin. Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming clock. Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming data.
RB1	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3/PGM	36	39	11	I/O	TTL	
RB4	37	41	14	I/O	TTL	
RB5	38	42	15	I/O	TTL	
RB6/PGC	39	43	16	I/O	TTL/ST ⁽²⁾	
RB7/PGD	40	44	17	I/O	TTL/ST ⁽²⁾	

Legend: I = input O = output I/O = input/output P = power
 -- = Not used TTL = TTL input ST = Schmitt Trigger input

- Note 1:** This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE 1-2: PIC16F874 AND PIC16F877 PINOUT DESCRIPTION (CONTINUED)

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
RC0/T1OSO/T1CKI	15	16	32	I/O	ST	PORTC is a bi-directional I/O port. RC0 can also be the Timer1 oscillator output or a Timer1 clock input. RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output. RC2 can also be the Capture1 input/Compare1 output/PWM1 output. RC3 can also be the synchronous serial clock input/output for both SPI and I ² C modes. RC4 can also be the SPI Data In (SPI mode) or data I/O (I ² C mode). RC5 can also be the SPI Data Out (SPI mode). RC6 can also be the USART Asynchronous Transmit or Synchronous Clock. RC7 can also be the USART Asynchronous Receive or Synchronous Data.
RC1/T1OSI/CCP2	16	18	35	I/O	ST	
RC2/CCP1	17	19	36	I/O	ST	
RC3/SCK/SCL	18	20	37	I/O	ST	
RC4/SDI/SDA	23	25	42	I/O	ST	
RC5/SDO	24	26	43	I/O	ST	
RC6/TX/CK	25	27	44	I/O	ST	
RC7/RX/DT	26	29	1	I/O	ST	
RD0/PSP0	19	21	38	I/O	ST/TTL ⁽³⁾	PORTD is a bi-directional I/O port or parallel slave port when interfacing to a microprocessor bus.
RD1/PSP1	20	22	39	I/O	ST/TTL ⁽³⁾	
RD2/PSP2	21	23	40	I/O	ST/TTL ⁽³⁾	
RD3/PSP3	22	24	41	I/O	ST/TTL ⁽³⁾	
RD4/PSP4	27	30	2	I/O	ST/TTL ⁽³⁾	
RD5/PSP5	28	31	3	I/O	ST/TTL ⁽³⁾	
RD6/PSP6	29	32	4	I/O	ST/TTL ⁽³⁾	
RD7/PSP7	30	33	5	I/O	ST/TTL ⁽³⁾	
RE0/RD $\overline{\text{AN}}5$	8	9	25	I/O	ST/TTL ⁽³⁾	PORTE is a bi-directional I/O port. RE0 can also be read control for the parallel slave port, or analog input5. RE1 can also be write control for the parallel slave port, or analog input6. RE2 can also be select control for the parallel slave port, or analog input7.
RE1/WR $\overline{\text{AN}}6$	9	10	26	I/O	ST/TTL ⁽³⁾	
RE2/CS $\overline{\text{AN}}7$	10	11	27	I/O	ST/TTL ⁽³⁾	
Vss	12,31	13,34	6,29	P	—	Ground reference for logic and I/O pins.
VDD	11,32	12,35	7,28	P	—	Positive supply for logic and I/O pins.
NC	—	1,17,28,40	12,13,33,34	—	—	These pins are not internally connected. These pins should be left unconnected.

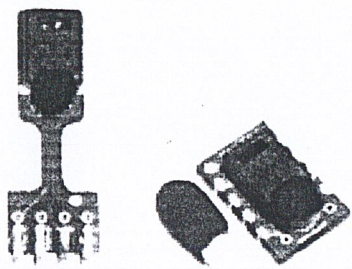
Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

- Note 1:** This buffer is a Schmitt Trigger input when configured as an external interrupt.
Note 2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
Note 3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
Note 4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SHT1x / SHT7x
Humidity & Temperature
Sensmitter

- _ Relative humidity and temperature sensors
- _ Dew point
- _ Fully calibrated, digital output
- _ No external components required
- _ Ultra low power consumption
- _ Surface mountable or 4-pin fully interchangeable
- _ Excellent long-term stability
- _ Small size
- _ Automatic power down



Preliminary Information June 2002

SHT1x Product Summary

The SHT1x / SHT7x is a single chip relative humidity and temperature multi sensor module comprising a calibrated digital output. Application of industrial CMOS processes with customized post processing (CMOSens® technology) ensures highest reliability and excellent long term stability. The device includes two calibrated microsensors for relative humidity and temperature which are seamlessly coupled to a 14bit analog to digital converter and a serial interface circuit on the same chip. This results in superior signal quality, a fast response time and insensitivity to external disturbances (EMC) at a very competitive price. Each sensor is calibrated in a precision humidity chamber and the calibration coefficients are programmed into the

OTP memory. These coefficients are used internally during measurements to calibrate the signals from the sensors. The 2-wire serial interface and internal voltage regulation allows easy and fast system integration. Its tiny size and low power consumption makes it the ultimate choice for even the most demanding applications including automotive, instrumentation, medical equipment, heating, ventilation and air conditioning systems (HVAC), portable consumer electronics and battery-operated controllers. The device is supplied in either a surface-mountable LCC (SHT1x) or as a 4-pin single-in-line type package (SHT7x). Customer specific packaging options may be available on request.

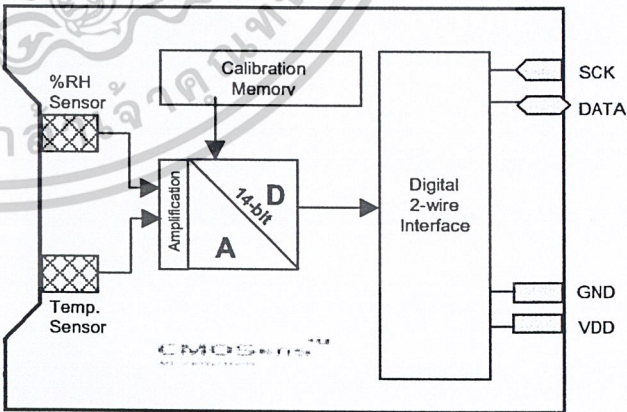
Applications

- _ Consumer Appliances
- _ Automotive
- _ HVAC
- _ Weather stations
- _ Test & Measurement
- _ Data Logging
- _ Automation
- _ White Goods

Ordering information

Part Number	Humidity accuracy	Temperature accuracy	Package	
SHT11	±3.5%RH	±0.5°C@25°C	SMT	
SHT15	±2.0%RH	±0.5°C	SMT	
SHT71	±3.5%RH	±0.5°C@25°C	4-pin single-in-line	Available Q3 2002
SHT75	±2.0%RH	±0.5°C	4-pin single-in-line	Available Q3 2002

Schematic Diagram



1 Sensor Performance Specifications⁽¹⁾

Parameter	Conditions	Min.	Typ.	Max.	Units
Humidity					
Resolution		0.5	0.03	0.03	% RH
		8	12	12	bit
Repeatability			±0.1		% RH
Accuracy ⁽²⁾ & Interchangeability		see figure 1			
Nonlinearity	10 - 90 %RH	<1 ⁽⁵⁾		±3 ⁽⁵⁾	% RH
Range		0		100	% RH
Response time	1/e (63%) slowly moving air		4		s
Hysteresis			±1		% RH
Long term stability	Typical		< 1		% RH/yr
Temperature					
Resolution		0.04	0.01	0.01	°C
		0.07	0.02	0.02	°F
		12	14	14	bit
Repeatability			±0.1		°C
			±0.2		°F
Accuracy		see figure 1			
Range		-40		123.8	°C
		-40		254.9	°F
Response Time	1/e (63%)	5		30	s

Table 1 Sensor Performance Specifications

1.1 Converting the digital output to physical values

1.1.1 Humidity

To compensate for the non-linearity of the humidity sensor and to obtain the full accuracy it is recommended to convert the readout with the following formula:

$$RH_{linear} = c_1 + c_2 \cdot SO_{RH} + c_3 \cdot SO_{RH}^2$$

$c_1 = -4 \quad c_2 = 0.0405 \quad c_3 = -2.8 \cdot 10^{-6} \quad \text{for 12bit } SO_{RH}$
 $c_1 = -4 \quad c_2 = 0.648 \quad c_3 = -7.2 \cdot 10^{-4} \quad \text{for 8bit } SO_{RH}$

For simplified, less computation intense conversion formulas see application note "RH Non-Linearity Compensation".

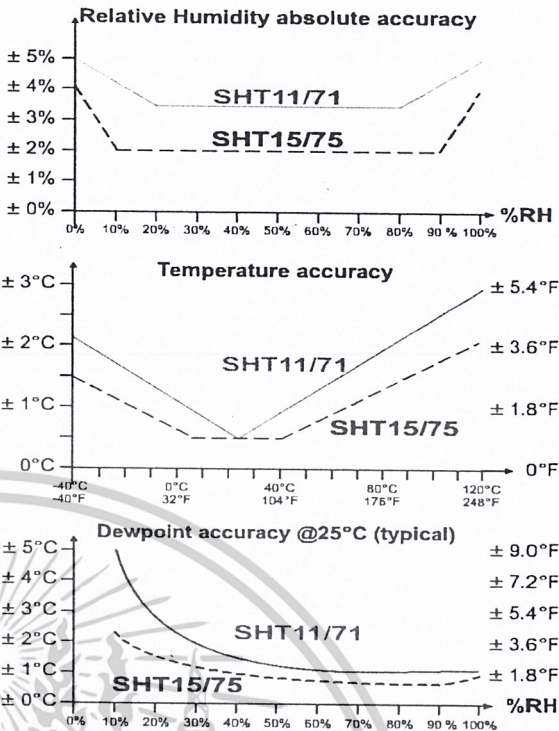
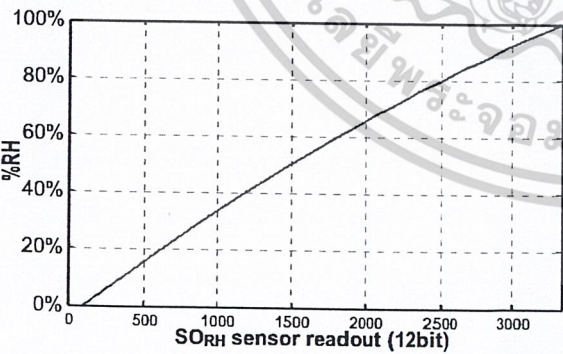


Figure 1 Rel. Humidity, Temperature and Dewpoint accuracies

For temperatures significantly different from 25°C (~77°F) the temperature coefficient of the RH sensor should be considered:

$$RH_{true} = (T_{°C} - 25) \cdot (t_1 + t_2 \cdot SO_{RH}) + RH_{linear}$$

with $t_1 = 0.01$; $t_2 = 0.00008$ for 12bit SO_{RH} ; $t_2 = 0.00128$ for 8bit SO_{RH}
This equals ~0.12%RH / °C @ 50%RH

1.1.2 Temperature

The temperature sensor is very linear by design. Use the following formula to convert from digital readout to temperature: Temperature = $d_1 + d_2 \cdot SO_T$
Use the appropriate table entries for 5V or 3V.

SO _T	Celsius		Fahrenheit	
	d ₁	d ₂	d ₁	d ₂
14bit 5V	-40	0.01	-40	0.018
12bit 5V	-40	0.04	-40	0.072
14bit 3V	-38.4	0.0098	-37.1	0.0176
12bit 3V	-38.4	0.0392	-37.1	0.0704

This equals a voltage dependency of ~ -0.2°C/V @ 25°C

1.1.3 Dewpoint

See application note "Dewpoint calculation" for more information.

⁽¹⁾ For operation within normal operation range as described in Chapter 3, RH accuracy at 25°C
⁽²⁾ Not including non-linearity
⁽³⁾ The default measurement resolution of 14bit (temperature) and 12bit (humidity) can be reduced to 12 and 8 bit through the status register.
⁽⁴⁾ Where SO_{RH} is the sensor output for relative humidity
⁽⁵⁾ Min. value after compensation with formula in chapter 1.1.1, Max. value without any compensation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2 Serial Interface

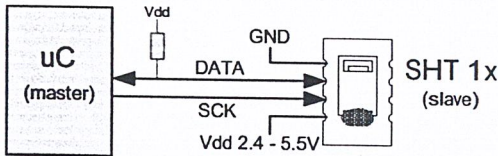


Figure 2 Typical application circuit

2.1 Power Pins

The device requires a voltage supply between 2.4V and 5.5V. After powerup the device requires 11ms to reach its "sleep" state. No commands should be sent before that time. Power supply pins (VDD, GND) may be decoupled with a 100 nF capacitor.

2.2 I/O Pins (Bidirectional 2-wire Interface)

See Table 5 for a detailed IO characteristics.

2.2.1 Serial clock input (SCK)

The SCK is used to synchronize the communication between a master and the SHT1x/SHT7x. Since the device contains fully static logic there is no minimum SCK frequency.

2.2.2 Serial data (DATA)

The DATA tristate pin is used to transfer data in and out of the device. DATA changes at the falling edge and is valid on the rising edge of the serial clock SCK. An external pull-up resistor is required to pull the signal high. (See Figure 2). Pull-up resistors are often included in I/O circuits of microcontrollers.

2.2.3 Command sequence

To initiate a transmission a "Transmission Start" sequence has to be issued. It consists of a lowering of the DATA line

while SCK is high, followed by a low pulse on SCK and raising DATA again while SCK is still high.

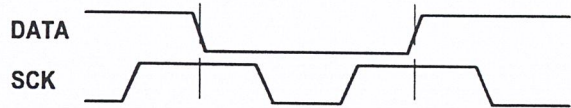


Figure 3 "Transmission Start" sequence

The subsequent command sequence consists of three address bits (only "000" is currently supported) and five command bits. The SHT1x/SHT7x indicates the proper reception of a command by pulling the DATA pin low (ACK bit) after the falling edge of the 8th SCK clock and the DATA line is released (and goes high) after the falling edge of the 9th SCK clock.

See 2.2.5 "Measurement Sequence" for an application of the command sequence

2.2.4 Connection reset sequence

If communication with the device is lost the following signal sequence will reset its serial interface:

While leaving DATA high toggle SCK 9 or more times. This must be followed by a "Transmission Start" sequence preceding the next command. This sequence resets the interface only. The status register preserves its content.

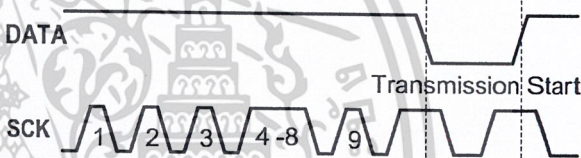
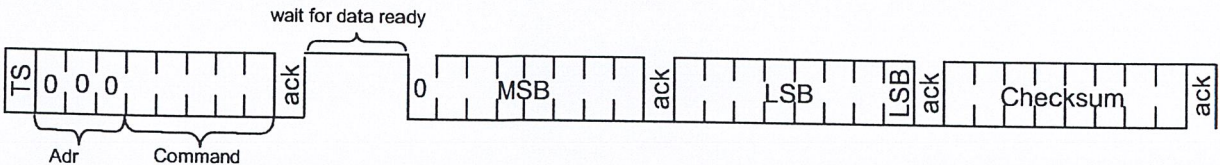


Figure 4 Connection reset sequence

Command	Code	Description
Reserved	0000x	Reserved
Measure Temperature	00011	Temperature measurement
Measure Humidity	00101	Humidity measurement
Status Register Read	00111	Read access to the status register (see application note)
Status Register Write	00110	Write access to the status register (see application note)
Reserved	0101x-1110x	Reserved
Soft reset	11110	resets the chip, clears the status register to default values wait 11ms before next command

Table 2 SHT1x/SHT7x list of commands



2.2.5 Measurement sequence (T and RH)

After issuing a measurement command ('00000101' for RH, '00000011' for Temperature) the controller has to wait for the measurement to complete. This takes approximately 11/55/210ms for a 8/12/14bit measurement. The exact time varies by up to $\pm 15\%$ with the speed of the internal oscillator. To signal the completion of a measurement, the SHT1x pulls down the data line (2) and the controller must restart SCK. Two bytes of measurement data and one byte of CRC checksum will then be transmitted. The uC must acknowledge each byte by pulling the DATA line low. All values are MSB first, right justified. (e.g. the 5th SCK is MSB for a 12bit value, for a 8bit result the first byte is not used). Communication terminates after the acknowledge bit of the

CRC data. If CRC-8 Checksum is not used the controller may terminate the communication after the measurement data LSB by keeping ack high.

The device automatically returns to sleep mode after the measurement and communication have finished.

Warning: To keep heat up of the SHT1x/SHT7x below 0.1°C it should not be active for more than 15% of the time (e.g. max. 3 measurements / second for 12bit accuracy).

2.2.6 CRC-8 Checksum Calculation

Please consult application note "CRC-8 Checksum Calculation" for information on how to calculate the CRC.

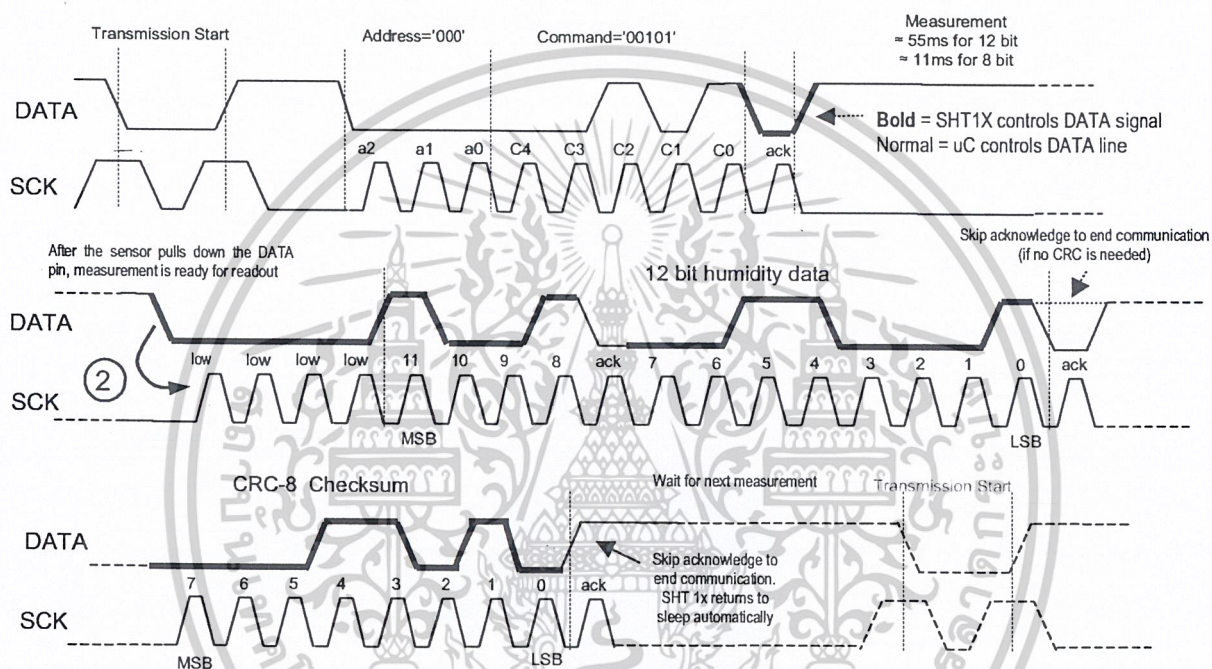


Figure 5 Example RH measurement sequence for value '0000'1001' 0011'0001' = 2353 = 75.79%RH

2.3 Status Register

Some of the advanced functions of the SHT1x/SHT7x are available through the status register. The following section gives a brief overview of these features. Please consult application note "Status Register" for more information.

2.3.1 Measurement resolution

The default measurement resolution of 14bit (temperature) and 12bit (humidity) can be reduced to 12 and 8 bit. This is especially useful in high speed or extreme low power applications.

2.3.2 End of Battery

The "End of Battery" function detects VDD voltages below 2.47V. Accuracy is $\pm 0.05V$

2.3.3 Heater

An on chip heating element can be switched on. It will increase the temperature of the sensor by approximately 5°C. Power consumption will increase by 8mA @ 5V.

Applications:

- By comparing temperature and humidity values before and after switching on the heater, proper functionality of both sensors can be verified.
- In high RH environments heating the sensor element will avoid condensation.

Warning: The built-in calibration is not correct while the sensmitter is heated!

Please consult application note "Status Register" for more information on how to access and use these features.

3 Specifications SHT1x/SHT7x

3.1 Absolute Maximum Ratings

Ambient Storage Temperature: -40°C to 120°C

3.2 Operating Conditions

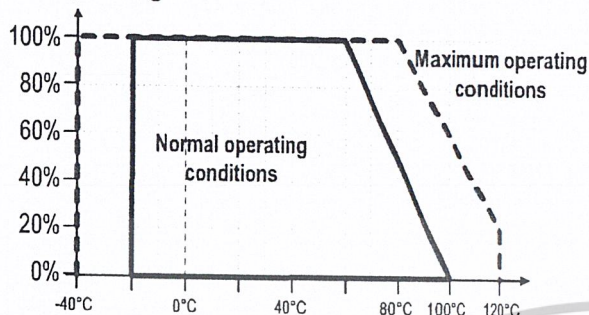


Figure 6 Recommended operating conditions

Conditions outside the recommended range may temporarily offset the RH signal up to $\pm 3\%$ RH. After return to normal conditions it will slowly return close to calibration state by itself.

To accelerate this process we recommend the following reconditioning procedure:

90°C at $< 5\%$ RH for 24h followed by
20-30°C at $> 74\%$ RH for 48h

Prolonged exposure to extreme conditions may accelerate ageing.

3.3 Special Conditions

Extensive tests were performed in various environments.

Environment	Norm	Results ⁽⁴⁾
Temperature Cycles	JESD22-A104-A -40 +125°C, 1000cy	Within Specifications
Pressure Cooker	JESD22-A110-B 2.3bar 125°C 85%RH	Reversible shift by +2% RH
Salt Atmosphere	DIN-50021ss	Within Specifications
Freezing cycles fully submerged	-20 +90°C, 100cy 30min dwell time	Reversible shift by +2% RH

Table 3 Qualification tests

Please contact SENSIRION for additional qualification information.

3.4 Electrical Specifications⁽¹⁾

3.4.1 ESD (Electrostatic Discharge)

ESD immunity is qualified according to MIL STD 883E, method 3015 (Human Body Model at $\pm 2\text{kV}$).

Latch-up immunity is provided at a force current of $\pm 100\text{ mA}$ with $T_{\text{amb}} = 80^\circ\text{C}$ according to JEDEC 17.

See application note "ESD, Latchup and EMC" for more information.

3.4.2 DC Characteristics

VDD=5V, Temperature= 25°C unless otherwise noted

Parameter	Conditions	Min.	Typ.	Max.	Units
Power supply DC		2.4	5	5.5	V
Supply current	measuring		550		μA
	average	2 ⁽²⁾	28 ⁽³⁾		μA
	sleep		0.3	1	μA
Low level output voltage		0		20% Vdd	Vdd
High level output voltage		75%		100%	Vdd
Low level input voltage	Negative going	0		20% Vdd	Vdd
High level input voltage	Positive going	80%		100%	Vdd
Input current on pads				1	μA
Output peak current	on			4	mA
	Tristated (off)		10		μA

Table 4 SHT1x/SHT7x DC Characteristics

3.4.3 I/O Characteristics

	Parameter	Conditions	Min	Typ.	Max.	Unit
F _{SCK}	SCK frequency	VDD > 4.5 V			10	MHz
		VDD < 4.5 V			1	MHz
T _{RFO}	DATA fall time	Output load 5 pF	3.5	10	20	ns
		Output load 100 pF	30	40	200	ns
T _{CLH}	SCK high time		100			ns
T _{CLL}	SCK low time		100			ns
T _V	DATA valid time			250		ns
T _{SU}	DATA set up time		100			ns
T _{HO}	DATA hold time		0	10		ns
T _{R/F}	SCK rise/fall time				200	ns

Table 5 SHT1x/SHT7x I/O Signals Characteristics

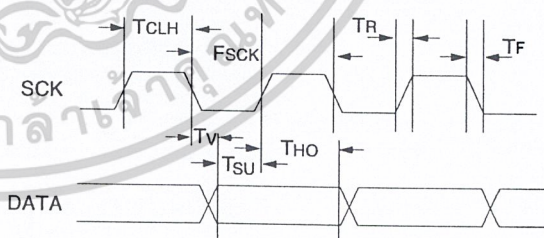


Figure 7 Timing Diagram

⁽¹⁾ Parameters are periodically sampled and not 100% tested

⁽²⁾ With one measurement of 8 bit accuracy without OTP reload per second

⁽³⁾ With one measurement of 12bit accuracy per second

⁽⁴⁾ The temperature sensor passed all tests without any drift Package and electronics also passed 100%

4 Physical Dimensions and Mounting Information

4.1 SHT1x (surface mountable)

Pin	Name	Comment
1	GND	Ground
2	DATA	Serial data bidirectional
3	SCK	Serial clock input
4	VDD	Supply 2.4 – 5.5V
5-8	nc	Do not connect pins on right side

Table 6 SHT1x Pin Description

4.1.1 Package type

The SHT1x is supplied in a surface-mountable LCC type package. The sensors housing consists of a Liquid Crystal Polymer (LCP) cap with epoxy glob top on a standard 0.8mm FR4 substrate.

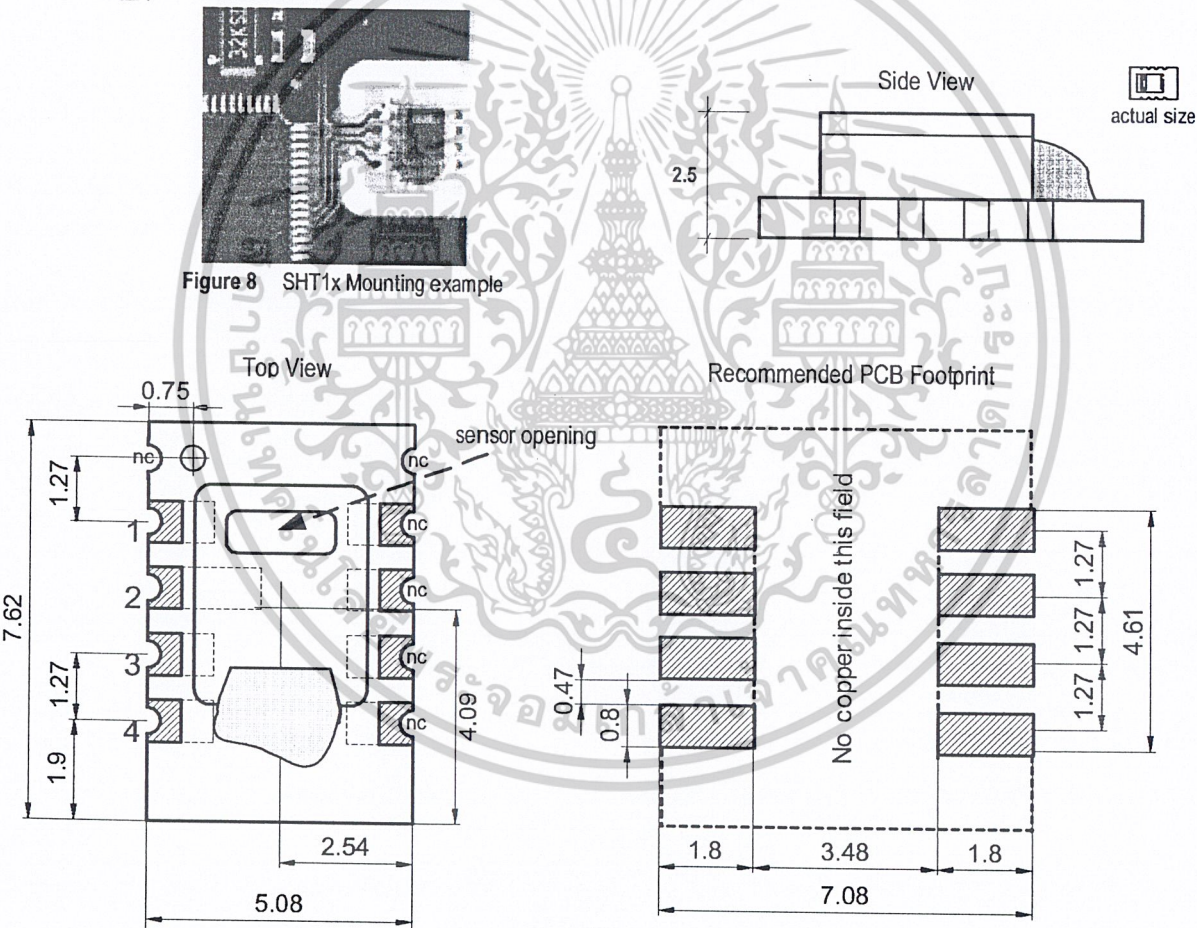
Device size is 7.62 x 5.08 x 2.5 mm. Weight 100mg

4.1.2 Soldering Information

The SHT1x can be soldered using standard reflow ovens at maximum 225°C for 20 seconds. For manual soldering contact time must be limited to 5 seconds at up to 350°C. After soldering the devices must be stored at >74%RH for at least 24h to allow the polymer to recover. Please consult the application note “Soldering procedure” for detailed instructions.

4.1.3 Delivery Conditions

The SHT1x will be delivered in standard IC tubes by 80 pieces per tube. Other delivery options may be available on request.



4.2 SHT7x (4-pin single-in-line)

SHT7x are available starting Q3 2002

Pin	Name	Comment
1	SCK	Serial clock input
2	VDD	Supply 2.4 – 5.5V
3	GND	Ground
4	DATA	Serial data bidirectional

Table 7 SHT7x Pin Description

4.2.1 Package type

The device is supplied in a single-in-line pin type package. The sensors housing consists of a Liquid Crystal Polymer (LCP) cap with epoxy glob top on a standard 0.6mm FR4 substrate.

The sensor head is connected to the pins by a small bridge to minimize heat conduction and response times.

A 100nF capacitor is mounted on the back side between VDD and GND.

Weight 168mg, Weight of sensor head 73mg

All pins are gold plated to avoid corrosion.

Pins mate with most 1.27mm (0.05") sockets

e.g.: Preci-dip / Mill-Max 851-93-004-20-001 or similar

4.2.2 Soldering Information (Preliminary)

The SHT7x may be soldered using standard wave soldering systems at maximum 225°C for 20 seconds.

For manual soldering contact time must be limited to 5 seconds at up to 350°C.

After soldering the devices must be stored at >74%RH for at least 24h to allow the polymer to recover.

Please consult the application note "Soldering procedure" for detailed instructions.

4.2.3 Delivery Conditions

The SHT7x will be delivered in trays by xx pieces per tray. Other delivery options may be available on request.

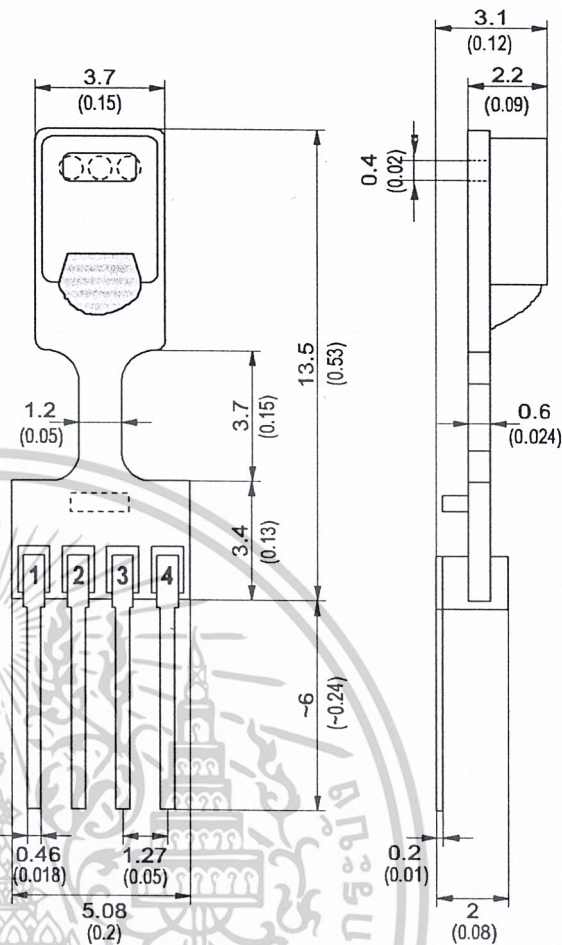


Figure 10 SHT7x dimensions in mm (inch), preliminary

4.3 Other Packages

Other packaging options may be available on request.

4.4 Production date

The production date is printed onto the cap in white numbers in the form wwyy. e.g. "351" = week 35, 2001.

4.5 Mounting Recommendations

The relative humidity of a gas strongly depends on its temperature. It is therefore essential to keep the sensor at the same temperature as the air of which the humidity is to be measured.

If the sensmitter shares a PCB with heating electronic components it should be mounted below the heat source and the housing must remain well ventilated. To reduce

heat conduction copper layers between the SHT1x and the rest of the PCB should be minimized and a slit may be milled in between.

Prolonged direct exposure of the SHT1x/SHT7x to sunshine or UV radiation should be avoided.

4.6 Wiring considerations and signal integrity

Carrying the SCK and DATA signal parallel and in close proximity (e.g. in wires) for more than 10cm may result in cross talk and loss of communication. This may be resolved by routing VDD and/or GND between the two data signals.

Please see the application note "ESD, Latchup and EMC" for more information.

5 Revision history

Date	Page	Changes
February 2002	1-9	First public release
February (2) 2002	4	Corrected CRC information to match application note
March 2002	2	Extended SHT11 3.5 accuracy range to 20%-80%
	8	Added image of mounting example
	2	Changed coefficients of temperature conversion formula
		Various small modifications
June 2002	6	Extended "no copper" area in figure 9
	5	Corrected timing diagram and table 5 with setup times for DATA
	5	Added qualification data table
	7	Added SHT7x information
	4	Renamed "End of Life" function to "End of Battery"

The latest version of this document and all application notes can be found at:
www.sensirion.com/en/download/humiditysensor/SHT11.htm

6 Important Notices

The warranty for each SENSIRION AG product comes in the form of a written warranty which governs sale and use of such product. Such warranty is contained in the printed terms and conditions under which such product is sold, or in a separate written warranty supplied with the product. Please refer to such written warranty with respect to its applicability to certain applications of such product.

These products may be subject to restrictions on use. Please contact SENSIRION AG for a list of the current additional restrictions on these products. By purchasing these products, the purchaser of these products agrees to comply with such restrictions. Please contact SENSIRION AG for clarification of any restrictions described herein. SENSIRION AG reserves the right, without further notice, to change the SENSIRION SHT1x/SHT7x Relative Humidity and Temperature Sensor product specifications and/or information in this document and to improve reliability, functions and design.

SENSIRION AG assumes no responsibility or liability for any use of SENSIRION SHT1x/SHT7x product. Application examples and alternative uses of the SENSIRION

SHT1x/SHT7x are for illustration purposes only and SENSIRION AG makes no representation or warranty that such applications shall be suitable for the use specified.

Copyright© 2001-2002, SENSIRION AG.
All rights reserved.

7 Caution

The inherent design of this component causes it to be sensitive to electrostatic discharge (ESD). To prevent ESD-induced damage and/or degradation, take normal ESD precautions when handling this product. See application note "ESD, Latchup and EMC" for more information.

8 Warning, personal injury

Do not use this product as safety or emergency stop devices or in any other application where failure of the product could result in personal injury. **Failure to comply with these instructions could result in death or serious injury.**

Headquarters and Sales Office

SENSIRION AG Phone: + 41 (0)1 306 40 00
Eggbühlstr. 14 Fax: + 41 (0)1 306 40 30
P.O. Box e-mail: info@sensirion.com
CH-8052 Zürich <http://www.sensirion.com/>
Switzerland

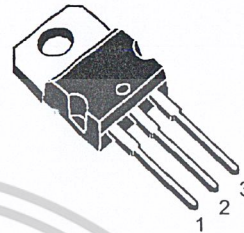
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPLEMENTARY SILICON POWER TRANSISTORS

■ SGS-THOMSON PREFERRED SALESTYPES

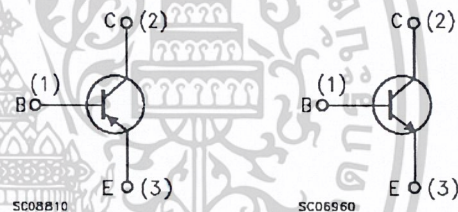
DESCRIPTION

The MJE3055T is a silicon epitaxial-base NPN transistor in Jedec TO-220 package. It is intended for power switching circuits and general-purpose amplifiers. The complementary PNP type is MJE2955T.



TO-220

INTERNAL SCHEMATIC DIAGRAM



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_{CEO}	Collector-Emitter Voltage ($I_B = 0$)	60	V
V_{CBO}	Collector-Base Voltage ($I_E = 0$)	70	V
V_{EBO}	Emitter-Base Voltage ($I_C = 0$)	5	V
I_C	Collector Current	10	A
I_B	Base Current	6	A
P_{tot}	Total Power Dissipation at $T_{case} \leq 25^\circ C$	75	W
T_{stg}	Storage Temperature	-55 to 150	$^\circ C$
T_J	Max. Operating Junction Temperature	150	$^\circ C$

For PNP types voltage and current values are negative.

THERMAL DATA

$R_{thj-case}$	Thermal Resistance Junction-case	Max	1.66	°C/W
----------------	----------------------------------	-----	------	------

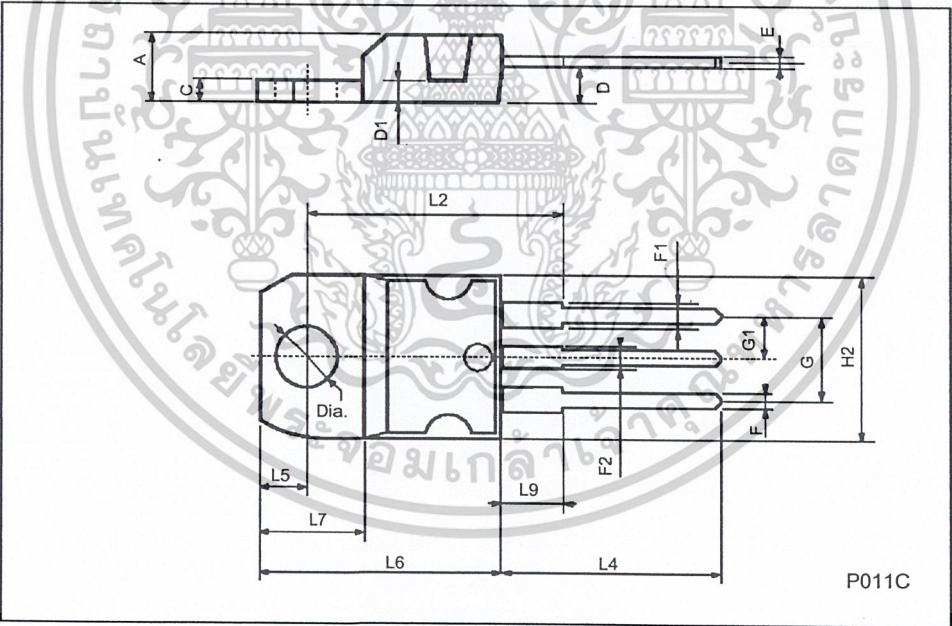
ELECTRICAL CHARACTERISTICS ($T_{case} = 25\text{ }^{\circ}\text{C}$ unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
I_{CEO}	Collector Cut-off Current ($I_B = 0$)	$V_{CE} = 30\text{ V}$			700	μA
I_{CEX}	Collector Cut-off Current ($V_{BE} = 1.5\text{ V}$)	$V_{CE} = 70\text{ V}$ $T_{CASE} = 150^{\circ}\text{C}$			1 5	mA mA
I_{CBO}	Collector Cut-off Current ($I_E = 0$)	$V_{CBO} = 70\text{ V}$ $T_{CASE} = 150^{\circ}\text{C}$			1 10	mA mA
I_{EBO}	Emitter Cut-off Current ($I_C = 0$)	$V_{EBO} = 5\text{ V}$			5	mA
$V_{CEO(sus)}^*$	Collector-Emitter Sustaining Voltage	$I_C = 200\text{ mA}$	60			V
$V_{CE(sat)}^*$	Collector-Emitter Sustaining Voltage	$I_C = 4\text{ A}$ $I_C = 10\text{ A}$	$I_B = 0.4\text{ A}$ $I_B = 3.3\text{ A}$		1.1 8	V V
$V_{BE(on)}^*$	Base-Emitter on Voltage	$I_C = 4\text{ A}$	$V_{CE} = 4\text{ V}$		1.8	V
h_{FE}	DC Current Gain	$I_C = 4\text{ A}$ $I_C = 10\text{ A}$	$V_{CE} = 4\text{ V}$ $V_{CE} = 4\text{ V}$	20 5	70	
f_T	Transistor Frequency	$I_C = 500\text{ mA}$ $f = 500\text{ KHz}$	$V_{CE} = 10\text{ V}$	2		MHz

* Pulsed: Pulse duration = 300 μs , duty cycle $\leq 2\%$
For PNP type voltage and current values are negative.

TO-220 MECHANICAL DATA

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A	4.40		4.60	0.173		0.181
C	1.23		1.32	0.048		0.051
D	2.40		2.72	0.094		0.107
D1		1.27			0.050	
E	0.49		0.70	0.019		0.027
F	0.61		0.88	0.024		0.034
F1	1.14		1.70	0.044		0.067
F2	1.14		1.70	0.044		0.067
G	4.95		5.15	0.194		0.203
G1	2.4		2.7	0.094		0.106
H2	10.0		10.40	0.393		0.409
L2		16.4			0.645	
L4	13.0		14.0	0.511		0.551
L5	2.65		2.95	0.104		0.116
L6	15.25		15.75	0.600		0.620
L7	6.2		6.6	0.244		0.260
L9	3.5		3.93	0.137		0.154
DIA.	3.75		3.85	0.147		0.151



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of SGS-THOMSON Microelectronics.

© 1995 SGS-THOMSON Microelectronics - All Rights Reserved

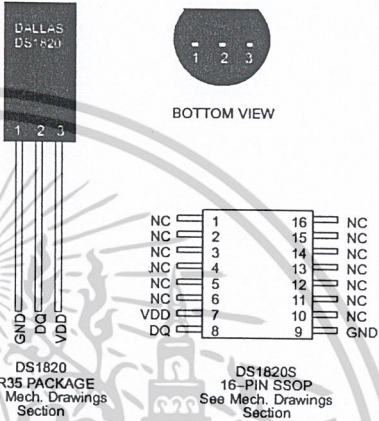
SGS-THOMSON Microelectronics GROUP OF COMPANIES
Australia - Brazil - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco - The Netherlands -
Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FEATURES

- Unique 1-Wire™ interface requires only one port pin for communication
- Multidrop capability simplifies distributed temperature sensing applications
- Requires no external components
- Can be powered from data line
- Zero standby power required
- Measures temperatures from -55°C to +125°C in 0.5°C increments. Fahrenheit equivalent is -67°F to +257°F in 0.9°F increments
- Temperature is read as a 9-bit digital value.
- Converts temperature to digital word in 200 ms (typ.)
- User-definable, nonvolatile temperature alarm settings
- Alarm search command identifies and addresses devices whose temperature is outside of programmed limits (temperature alarm condition)
- Applications include thermostatic controls, industrial systems, consumer products, thermometers, or any thermally sensitive system

PIN ASSIGNMENT



PIN DESCRIPTION

- GND – Ground
- DQ – Data In/Out
- VDD – Optional V_{DD}
- NC – No Connect

DESCRIPTION

The DS1820 Digital Thermometer provides 9-bit temperature readings which indicate the temperature of the device.

Information is sent to/from the DS1820 over a 1-Wire interface, so that only one wire (and ground) needs to be connected from a central microprocessor to a DS1820. Power for reading, writing, and performing temperature conversions can be derived from the data line itself with no need for an external power source.

Because each DS1820 contains a unique silicon serial number, multiple DS1820s can exist on the same 1-Wire bus. This allows for placing temperature sensors in many different places. Applications where this feature is useful include HVAC environmental controls, sensing temperatures inside buildings, equipment or machinery, and in process monitoring and control.

DETAILED PIN DESCRIPTION

PIN 16-PIN SSOP	PIN PR35	SYMBOL	DESCRIPTION
9	1	GND	Ground.
8	2	DQ	Data Input/Output pin. For 1-Wire operation: Open drain. (See "Parasite Power" section.)
7	3	V _{DD}	Optional V _{DD} pin. See "Parasite Power" section for details of connection.

DS1820S (16-pin SSOP): All pins not specified in this table are not to be connected.

OVERVIEW

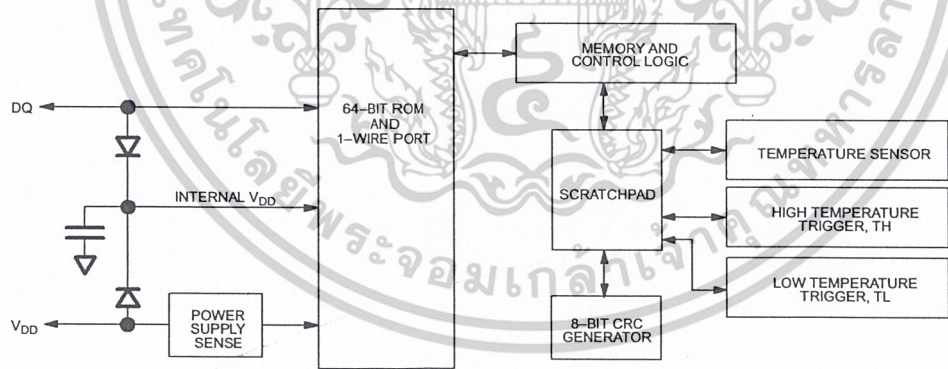
The block diagram of Figure 1 shows the major components of the DS1820. The DS1820 has three main data components: 1) 64-bit lasered ROM, 2) temperature sensor, and 3) nonvolatile temperature alarm triggers TH and TL. The device derives its power from the 1-Wire communication line by storing energy on an internal capacitor during periods of time when the signal line is high and continues to operate off this power source during the low times of the 1-Wire line until it returns high to replenish the parasite (capacitor) supply. As an alternative, the DS1820 may also be powered from an external 5 volts supply.

Communication to the DS1820 is via a 1-Wire port. With the 1-Wire port, the memory and control functions will not be available before the ROM function protocol has been established. The master must first provide one of five ROM function commands: 1) Read ROM, 2) Match ROM, 3) Search ROM, 4) Skip ROM, or 5) Alarm Search. These commands operate on the 64-bit lasered ROM portion of each device and can single out

a specific device if many are present on the 1-Wire line as well as indicate to the Bus Master how many and what types of devices are present. After a ROM function sequence has been successfully executed, the memory and control functions are accessible and the master may then provide any one of the six memory and control function commands.

One control function command instructs the DS1820 to perform a temperature measurement. The result of this measurement will be placed in the DS1820's scratchpad memory, and may be read by issuing a memory function command which reads the contents of the scratchpad memory. The temperature alarm triggers TH and TL consist of one byte EEPROM each. If the alarm search command is not applied to the DS1820, these registers may be used as general purpose user memory. Writing TH and TL is done using a memory function command. Read access to these registers is through the scratchpad. All data is read and written least significant bit first.

DS1820 BLOCK DIAGRAM Figure 1



PARASITE POWER

The block diagram (Figure 1) shows the parasite powered circuitry. This circuitry "steals" power whenever the I/O or V_{DD} pins are high. I/O will provide sufficient power as long as the specified timing and voltage requirements are met (see the section titled "1-Wire Bus System"). The advantages of parasite power are two-fold: 1) by parasiting off this pin, no local power source is needed for remote sensing of temperature, and 2) the ROM may be read in absence of normal power.

In order for the DS1820 to be able to perform accurate temperature conversions, sufficient power must be provided over the I/O line when a temperature conversion is taking place. Since the operating current of the DS1820 is up to 1 mA, the I/O line will not have sufficient drive due to the 5K pull-up resistor. This problem is particularly acute if several DS1820's are on the same I/O and attempting to convert simultaneously.

There are two ways to assure that the DS1820 has sufficient supply current during its active conversion cycle. The first is to provide a strong pull-up on the I/O line whenever temperature conversions or copies to the E² memory are taking place. This may be accomplished by using a MOSFET to pull the I/O line directly to the power supply as shown in Figure 2. The I/O line must be switched over to the strong pull-up within 10 μs maximum after issuing any protocol that involves copying to the E² memory or initiates temperature conversions. When using the parasite power mode, the V_{DD} pin must be tied to ground.

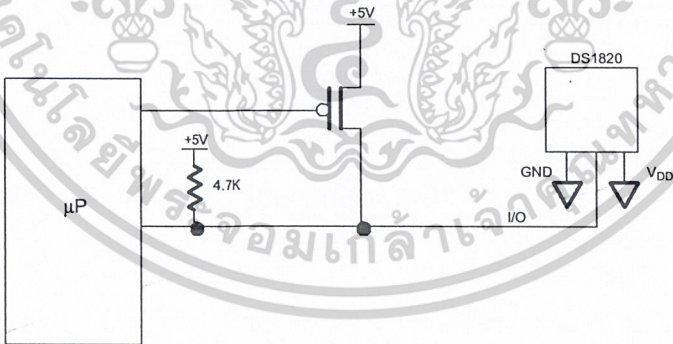
Another method of supplying current to the DS1820 is through the use of an external power supply tied to the

V_{DD} pin, as shown in Figure 3. The advantage to this is that the strong pull-up is not required on the I/O line, and the bus master need not be tied up holding that line high during temperature conversions. This allows other data traffic on the 1-Wire bus during the conversion time. In addition, any number of DS1820's may be placed on the 1-Wire bus, and if they all use external power, they may all simultaneously perform temperature conversions by issuing the Skip ROM command and then issuing the Convert T command. Note that as long as the external power supply is active, the GND pin may not be floating.

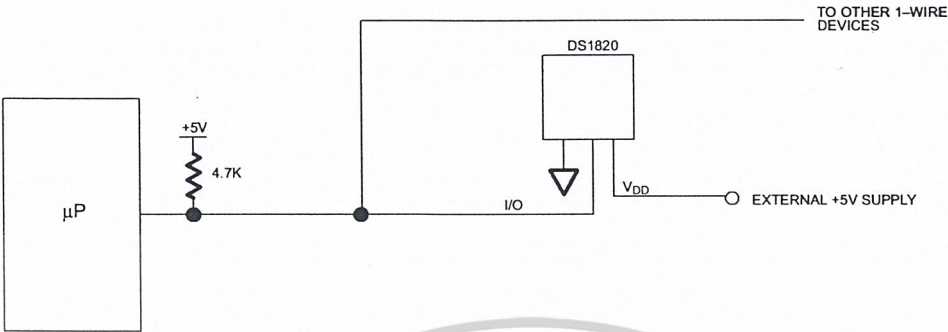
The use of parasite power is not recommended above 100°C, since it may not be able to sustain communications given the higher leakage currents the DS1820 exhibits at these temperatures. For applications in which such temperatures are likely, it is strongly recommended that V_{DD} be applied to the DS1820.

For situations where the bus master does not know whether the DS1820's on the bus are parasite powered or supplied with external V_{DD}, a provision is made in the DS1820 to signal the power supply scheme used. The bus master can determine if any DS1820's are on the bus which require the strong pull-up by sending a Skip ROM protocol, then issuing the read power supply command. After this command is issued, the master then issues read time slots. The DS1820 will send back "0" on the 1-Wire bus if it is parasite powered; it will send back a "1" if it is powered from the V_{DD} pin. If the master receives a "0", it knows that it must supply the strong pull-up on the I/O line during temperature conversions. See "Memory Command Functions" section for more detail on this command protocol.

STRONG PULL-UP FOR SUPPLYING DS1820 DURING TEMPERATURE CONVERSION Figure 2



USING V_{DD} TO SUPPLY TEMPERATURE CONVERSION CURRENT Figure 3



OPERATION – MEASURING TEMPERATURE

The DS1820 measures temperature through the use of an on-board proprietary temperature measurement technique. A block diagram of the temperature measurement circuitry is shown in Figure 4.

The DS1820 measures temperature by counting the number of clock cycles that an oscillator with a low temperature coefficient goes through during a gate period determined by a high temperature coefficient oscillator. The counter is preset with a base count that corresponds to -55°C. If the counter reaches zero before the gate period is over, the temperature register, which is also preset to the -55°C value, is incremented, indicating that the temperature is higher than -55°C.

At the same time, the counter is then preset with a value determined by the slope accumulator circuitry. This circuitry is needed to compensate for the parabolic behavior of the oscillators over temperature. The counter is then clocked again until it reaches zero. If the gate period is still not finished, then this process repeats.

The slope accumulator is used to compensate for the non-linear behavior of the oscillators over temperature, yielding a high resolution temperature measurement. This is done by changing the number of counts necessary for the counter to go through for each incremental degree in temperature. To obtain the desired resolution, therefore, both the value of the counter and the number of counts per degree C (the value of the slope accumulator) at a given temperature must be known.

Internally, this calculation is done inside the DS1820 to provide 0.5°C resolution. The temperature reading is

provided in a 16-bit, sign-extended two's complement reading. Table 1 describes the exact relationship of output data to measured temperature. The data is transmitted serially over the 1-Wire interface. The DS1820 can measure temperature over the range of -55°C to +125°C in 0.5°C increments. For Fahrenheit usage, a lookup table or conversion factor must be used.

Note that temperature is represented in the DS1820 in terms of a 1/2°C LSB, yielding the following 9-bit format:

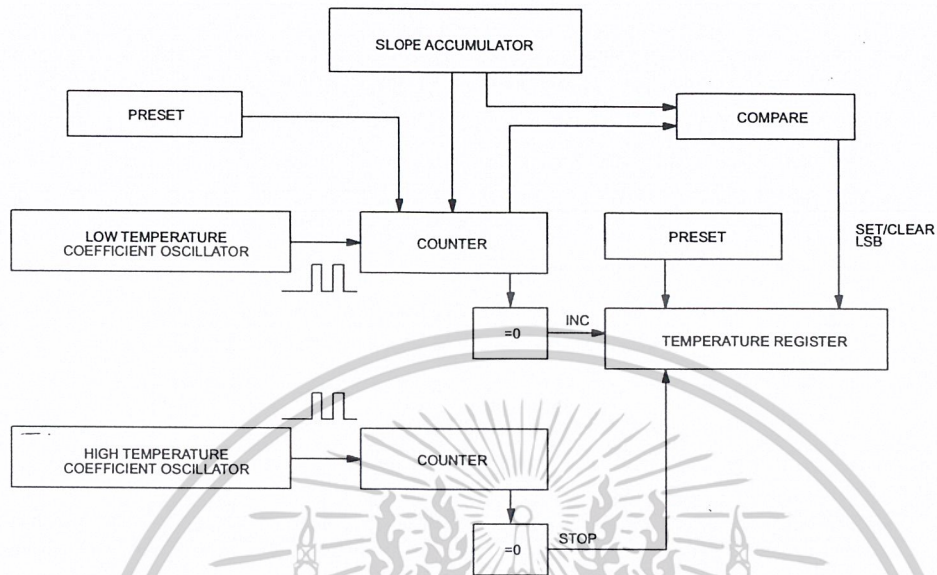
MSB								LSB
1	1	1	0	0	1	1	1	0
= -25°C								

The most significant (sign) bit is duplicated into all of the bits in the upper MSB of the two-byte temperature register in memory. This "sign-extension" yields the 16-bit temperature readings as shown in Table 1.

Higher resolutions may be obtained by the following procedure. First, read the temperature, and truncate the 0.5°C bit (the LSB) from the read value. This value is TEMP_READ. The value left in the counter may then be read. This value is the count remaining (COUNT_REMAIN) after the gate period has ceased. The last value needed is the number of counts per degree C (COUNT_PER_C) at that temperature. The actual temperature may then be calculated by the user using the following:

$$\text{TEMPERATURE} = \text{TEMP_READ} - 0.25 + \frac{(\text{COUNT_PER_C} - \text{COUNT_REMAIN})}{\text{COUNT_PER_C}}$$

TEMPERATURE MEASURING CIRCUITRY Figure 4



TEMPERATURE/DATA RELATIONSHIPS Table 1

TEMPERATURE	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+125°C	00000000 11111010	00FA
+25°C	00000000 00110010	0032h
+1/2°C	00000000 00000001	0001h
+0°C	00000000 00000000	0000h
-1/2°C	11111111 11111111	FFFFh
-25°C	11111111 11001110	FFCEh
-55°C	11111111 10010010	FF92h

OPERATION – ALARM SIGNALING

After the DS1820 has performed a temperature conversion, the temperature value is compared to the trigger values stored in TH and TL. Since these registers are 8-bit only, the 0.5°C bit is ignored for comparison. The most significant bit of TH or TL directly corresponds to the sign bit of the 16-bit temperature register. If the result of a temperature measurement is higher than TH or lower than TL, an alarm flag inside the device is set.

This flag is updated with every temperature measurement. As long as the alarm flag is set, the DS1820 will respond to the alarm search command. This allows many DS1820s to be connected in parallel doing simultaneous temperature measurements. If somewhere the temperature exceeds the limits, the alarming device(s) can be identified and read immediately without having to read non-alarming devices.

64-BIT LASERED ROM

Each DS1820 contains a unique ROM code that is 64-bits long. The first eight bits are a 1-Wire family code (DS1820 code is 10h). The next 48 bits are a unique serial number. The last eight bits are a CRC of the first 56 bits. (See Figure 5.) The 64-bit ROM and ROM Function Control section allow the DS1820 to operate as a 1-Wire device and follow the 1-Wire protocol detailed in the section "1-Wire Bus System". The functions required to control sections of the DS1820 are not accessible until the ROM function protocol has been satisfied. This protocol is described in the ROM function protocol flowchart (Figure 6). The 1-Wire bus master must first provide one of five ROM function commands: 1) Read ROM, 2) Match ROM, 3) Search ROM, 4) Skip ROM, or 5) Alarm Search. After a ROM functions sequence has been successfully executed, the functions specific to the DS1820 are accessible and the bus master may then provide and one of the six memory and control function commands.

CRC GENERATION

The DS1820 has an 8-bit CRC stored in the most significant byte of the 64-bit ROM. The bus master can compute a CRC value from the first 56-bits of the 64-bit ROM and compare it to the value stored within the DS1820 to determine if the ROM data has been received error-free by the bus master. The equivalent polynomial function of this CRC is:

CRC = X⁸ + X⁵ + X⁴ + 1

The DS1820 also generates an 8-bit CRC value using the same polynomial function shown above and pro-

vides this value to the bus master to validate the transfer of data bytes. In each case where a CRC is used for data transfer validation, the bus master must calculate a CRC value using the polynomial function given above and compare the calculated value to either the 8-bit CRC value stored in the 64-bit ROM portion of the DS1820 (for ROM reads) or the 8-bit CRC value computed within the DS1820 (which is read as a ninth byte when the scratchpad is read). The comparison of CRC values and decision to continue with an operation are determined entirely by the bus master. There is no circuitry inside the DS1820 that prevents a command sequence from proceeding if the CRC stored in or calculated by the DS1820 does not match the value generated by the bus master.

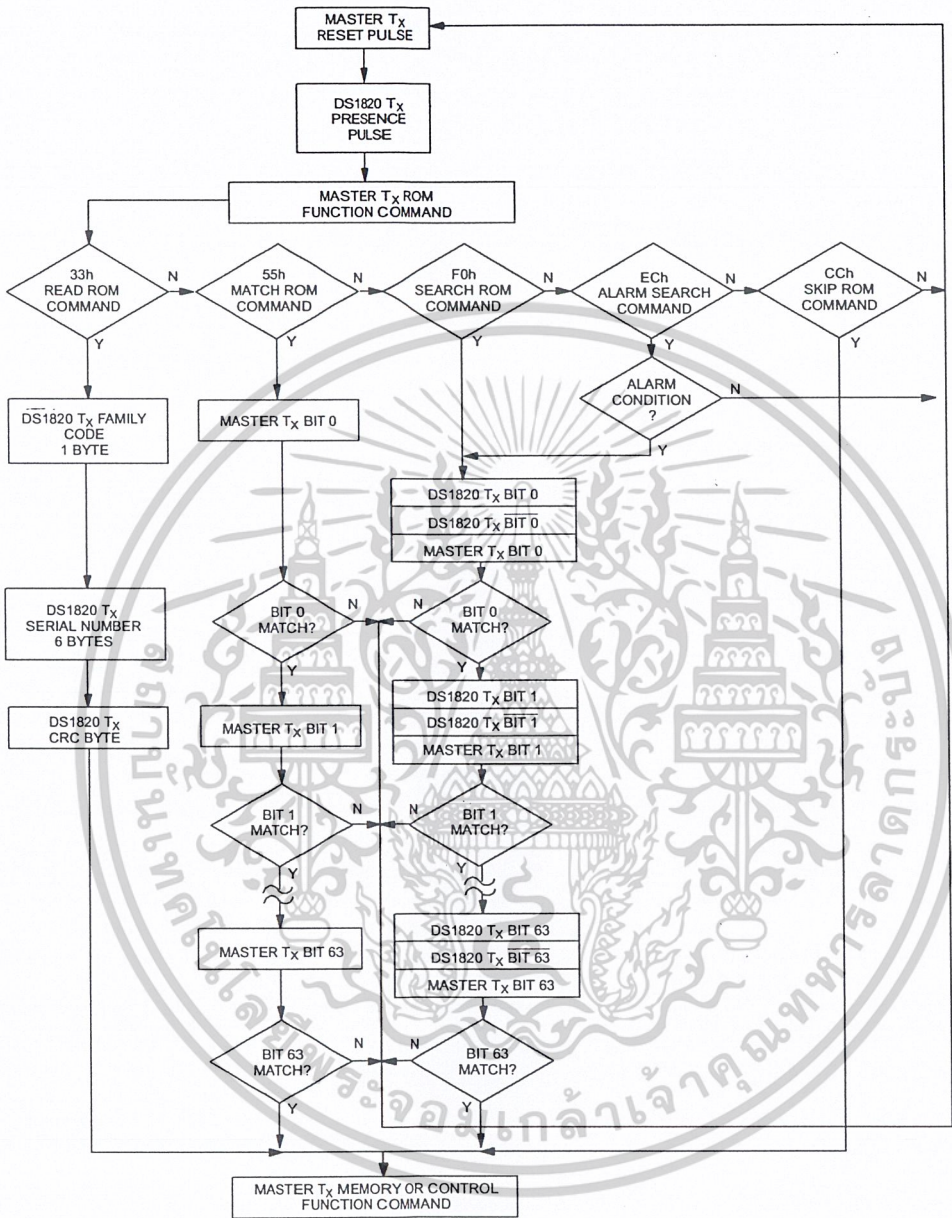
The 1-Wire CRC can be generated using a polynomial generator consisting of a shift register and XOR gates as shown in Figure 7. Additional information about the Dallas 1-Wire Cyclic Redundancy Check is available in Application Note 27 entitled "Understanding and Using Cyclic Redundancy Checks with Dallas Semiconductor Touch Memory Products".

The shift register bits are initialized to zero. Then starting with the least significant bit of the family code, one bit at a time is shifted in. After the 8th bit of the family code has been entered, then the serial number is entered. After the 48th bit of the serial number has been entered, the shift register contains the CRC value. Shifting in the eight bits of CRC should return the shift register to all zeros.

64-BIT LASERED ROM Figure 5

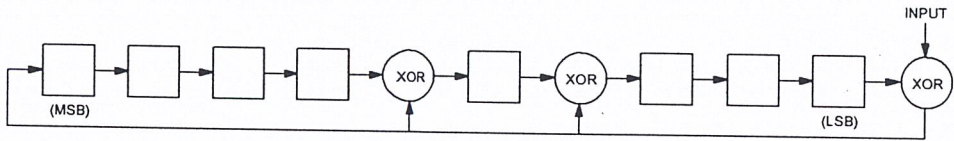
8-BIT CRC CODE		48-BIT SERIAL NUMBER		8-BIT FAMILY CODE (10h)	
MSB	LSB	MSB	LSB	MSB	LSB

ROM FUNCTIONS FLOW CHART Figure 6



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1-WIRE CRC CODE Figure 7



MEMORY

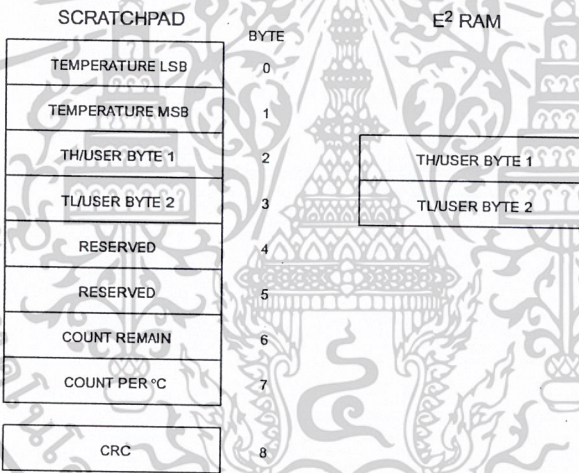
The DS1820's memory is organized as shown in Figure 8. The memory consists of a scratchpad RAM and a nonvolatile, electrically erasable (E²) RAM, which stores the high and low temperature triggers TH and TL. The scratchpad helps insure data integrity when communicating over the 1-Wire bus. Data is first written to the scratchpad where it can be read back. After the data has been verified, a copy scratchpad command will transfer the data to the nonvolatile (E²) RAM. This process insures data integrity when modifying the memory.

The scratchpad is organized as eight bytes of memory. The first two bytes contain the measured temperature

information. The third and fourth bytes are volatile copies of TH and TL and are refreshed with every power-on reset. The next two bytes are not used; upon reading back, however, they will appear as all logic 1's. The seventh and eighth bytes are count registers, which may be used in obtaining higher temperature resolution (see "Operation—measuring Temperature" section).

There is a ninth byte which may be read with a Read Scratchpad command. This byte contains a cyclic redundancy check (CRC) byte which is the CRC over all of the eight previous bytes. This CRC is implemented in the fashion described in the section titled "CRC Generation".

DS1820 MEMORY MAP Figure 8



1-WIRE BUS SYSTEM

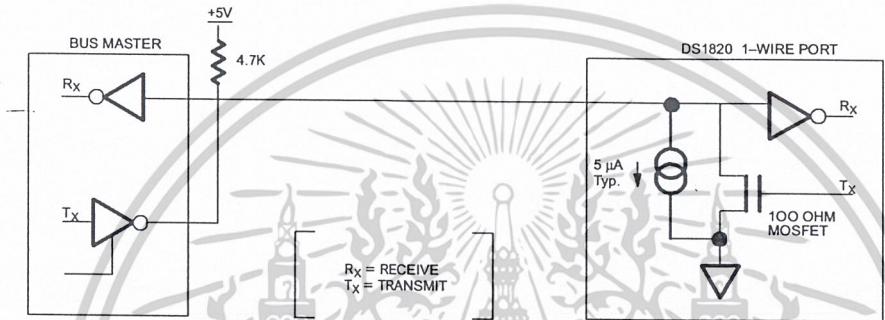
The 1-Wire bus is a system which has a single bus master and one or more slaves. The DS1820 behaves as a slave. The discussion of this bus system is broken down into three topics: hardware configuration, transaction sequence, and 1-Wire signaling (signal types and timing).

HARDWARE CONFIGURATION

The 1-Wire bus has only a single line by definition; it is important that each device on the bus be able to drive it

at the appropriate time. To facilitate this, each device attached to the 1-Wire bus must have open drain or 3-state outputs. The 1-Wire port of the DS1820 (I/O pin) is open drain with an internal circuit equivalent to that shown in Figure 9. A multidrop bus consists of a 1-Wire bus with multiple slaves attached. The 1-Wire bus requires a pullup resistor of approximately 5KΩ.

HARDWARE CONFIGURATION Figure 9



The idle state for the 1-Wire bus is high. If for any reason a transaction needs to be suspended, the bus MUST be left in the idle state if the transaction is to resume. Infinite recovery time can occur between bits so long as the 1-Wire bus is in the inactive (high) state during the recovery period. If this does not occur and the bus is left low for more than 480 μs, all components on the bus will be reset.

TRANSACTION SEQUENCE

The protocol for accessing the DS1820 via the 1-Wire port is as follows:

- Initialization
- ROM Function Command
- Memory Function Command
- Transaction/Data

INITIALIZATION

All transactions on the 1-Wire bus begin with an initialization sequence. The initialization sequence consists of a reset pulse transmitted by the bus master followed by presence pulse(s) transmitted by the slave(s).

The presence pulse lets the bus master know that the DS1820 is on the bus and is ready to operate. For more details, see the "1-Wire Signaling" section.

ROM FUNCTION COMMANDS

Once the bus master has detected a presence, it can issue one of the five ROM function commands. All ROM function commands are 8-bits long. A list of these commands follows (refer to flowchart in Figure 6):

Read ROM [33h]

This command allows the bus master to read the DS1820's 8-bit family code, unique 48-bit serial number, and 8-bit CRC. This command can only be used if there is a single DS1820 on the bus. If more than one slave is present on the bus, a data collision will occur when all slaves try to transmit at the same time (open drain will produce a wired AND result).

Match ROM [55h]

The match ROM command, followed by a 64-bit ROM sequence, allows the bus master to address a specific DS1820 on a multidrop bus. Only the DS1820 that exactly matches the 64-bit ROM sequence will respond to the following memory function command. All slaves that do not match the 64-bit ROM sequence will wait for a reset pulse. This command can be used with a single or multiple devices on the bus.

Skip ROM [CCh]

This command can save time in a single drop bus system by allowing the bus master to access the memory functions without providing the 64-bit ROM code. If more than one slave is present on the bus and a read command is issued following the Skip ROM command, data collision will occur on the bus as multiple slaves transmit simultaneously (open drain pullups will produce a wired AND result).

Search ROM [F0h]

When a system is initially brought up, the bus master might not know the number of devices on the 1-Wire bus or their 64-bit ROM codes. The search ROM command allows the bus master to use a process of elimination to identify the 64-bit ROM codes of all slave devices on the bus.

Alarm Search [ECh]

The flowchart of this command is identical to the Search ROM command. However, the DS1820 will respond to this command only if an alarm condition has been encountered at the last temperature measurement. An alarm condition is defined as a temperature higher than TH or lower than TL. The alarm condition remains set as long as the DS1820 is powered up, or until another temperature measurement reveals a non-alarming value. For alarming, the trigger values stored in EEPROM are taken into account. If an alarm condition exists and the TH or TL settings are changed, another temperature

conversion should be done to validate any alarm conditions.

Example of a ROM Search

The ROM search process is the repetition of a simple 3-step routine: read a bit, read the complement of the bit, then write the desired value of that bit. The bus master performs this simple, 3-step routine on each bit of the ROM. After one complete pass, the bus master knows the contents of the ROM in one device. The remaining number of devices and their ROM codes may be identified by additional passes.

The following example of the ROM search process assumes four different devices are connected to the same 1-Wire bus. The ROM data of the four devices is as shown:

ROM1	00110101...
ROM2	10101010...
ROM3	11110101...
ROM4	00010001...

The search process is as follows:

1. The bus master begins the initialization sequence by issuing a reset pulse. The slave devices respond by issuing simultaneous presence pulses.
2. The bus master will then issue the Search ROM command on the 1-Wire bus.
3. The bus master reads a bit from the 1-Wire bus. Each device will respond by placing the value of the first bit of their respective ROM data onto the 1-Wire bus. ROM1 and ROM4 will place a 0 onto the 1-Wire bus, i.e., pull it low. ROM2 and ROM3 will place a 1 onto the 1-Wire bus by allowing the line to stay high. The result is the logical AND of all devices on the line, therefore the bus master sees a 0. The bus master reads another bit. Since the Search ROM data command is being executed, all of the devices on the 1-Wire bus respond to this second read by placing the complement of the first bit of their respective ROM data onto the 1-Wire bus. ROM1 and ROM4 will place a 1 onto the 1-Wire, allowing the line to stay high. ROM2 and ROM3 will place a 0 onto the 1-Wire, thus it will be pulled low. The bus master again observes a 0 for the complement of the first ROM data bit. The bus master has determined that there are some devices on the 1-Wire bus that have a 0 in the first position and others that have a 1.

The data obtained from the two reads of the 3-step routine have the following interpretations:

- 00 There are still devices attached which have conflicting bits in this position.
 - 01 All devices still coupled have a 0-bit in this bit position.
 - 10 All devices still coupled have a 1-bit in this bit position.
 - 11 There are no devices attached to the 1-Wire bus.
4. The bus master writes a 0. This deselects ROM2 and ROM3 for the remainder of this search pass, leaving only ROM1 and ROM4 connected to the 1-Wire bus.
 5. The bus master performs two more reads and receives a 0-bit followed by a 1-bit. This indicates that all devices still coupled to the bus have 0's as their second ROM data bit.
 6. The bus master then writes a 0 to keep both ROM1 and ROM4 coupled.
 7. The bus master executes two reads and receives two 0-bits. This indicates that both 1-bits and 0-bits exist as the third bit of the ROM data of the attached devices.
 8. The bus master writes a 0-bit. This deselects ROM1 leaving ROM4 as the only device still connected.
 9. The bus master reads the remainder of the ROM bits for ROM4 and continues to access the part if desired. This completes the first pass and uniquely identifies one part on the 1-Wire bus.
 10. The bus master starts a new ROM search sequence by repeating steps 1 through 7.
 11. The bus master writes a 1-bit. This decouples ROM4, leaving only ROM1 still coupled.
 12. The bus master reads the remainder of the ROM bits for ROM1 and communicates to the underlying logic if desired. This completes the second ROM search pass, in which another of the ROMs was found.
 13. The bus master starts a new ROM search by repeating steps 1 through 3.
 14. The bus master writes a 1-bit. This deselects ROM1 and ROM4 for the remainder of this search pass, leaving only ROM2 and ROM3 coupled to the system.

15. The bus master executes two read time slots and receives two zeros.

16. The bus master writes a 0-bit. This decouples ROM3, and leaving only ROM2.

17. The bus master reads the remainder of the ROM bits for ROM2 and communicates to the underlying logic if desired. This completes the third ROM search pass, in which another of the ROMs was found.

18. The bus master starts a new ROM search by repeating steps 13 through 15.

19. The bus master writes a 1-bit. This decouples ROM2, leaving only ROM3.

20. The bus master reads the remainder of the ROM bits for ROM3 and communicates to the underlying logic if desired. This completes the fourth ROM search pass, in which another of the ROMs was found.

Note the following:

The bus master learns the unique ID number (ROM data pattern) of one 1-Wire device on each ROM Search operation. The time required to derive the part's unique ROM code is:

$$960 \mu\text{s} + (8 + 3 \times 64) 61 \mu\text{s} = 13.16 \text{ ms}$$

The bus master is therefore capable of identifying 75 different 1-Wire devices per second.

I/O SIGNALING

The DS1820 requires strict protocols to insure data integrity. The protocol consists of several types of signaling on one line: reset pulse, presence pulse, write 0, write 1, read 0, and read 1. All of these signals, with the exception of the presence pulse, are initiated by the bus master.

The initialization sequence required to begin any communication with the DS1820 is shown in Figure 11. A reset pulse followed by a presence pulse indicates the DS1820 is ready to send or receive data given the correct ROM command and memory function command.

The bus master transmits (TX) a reset pulse (a low signal for a minimum of 480 μs). The bus master then releases the line and goes into a receive mode (RX). The 1-Wire bus is pulled to a high state via the 5K pull-up resistor. After detecting the rising edge on the

I/O pin, the DS1820 waits 15–60 μ s and then transmits the presence pulse (a low signal for 60–240 μ s).

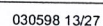
MEMORY COMMAND FUNCTIONS

The following command protocols are summarized in Table 2, and by the flowchart of Figure 10.

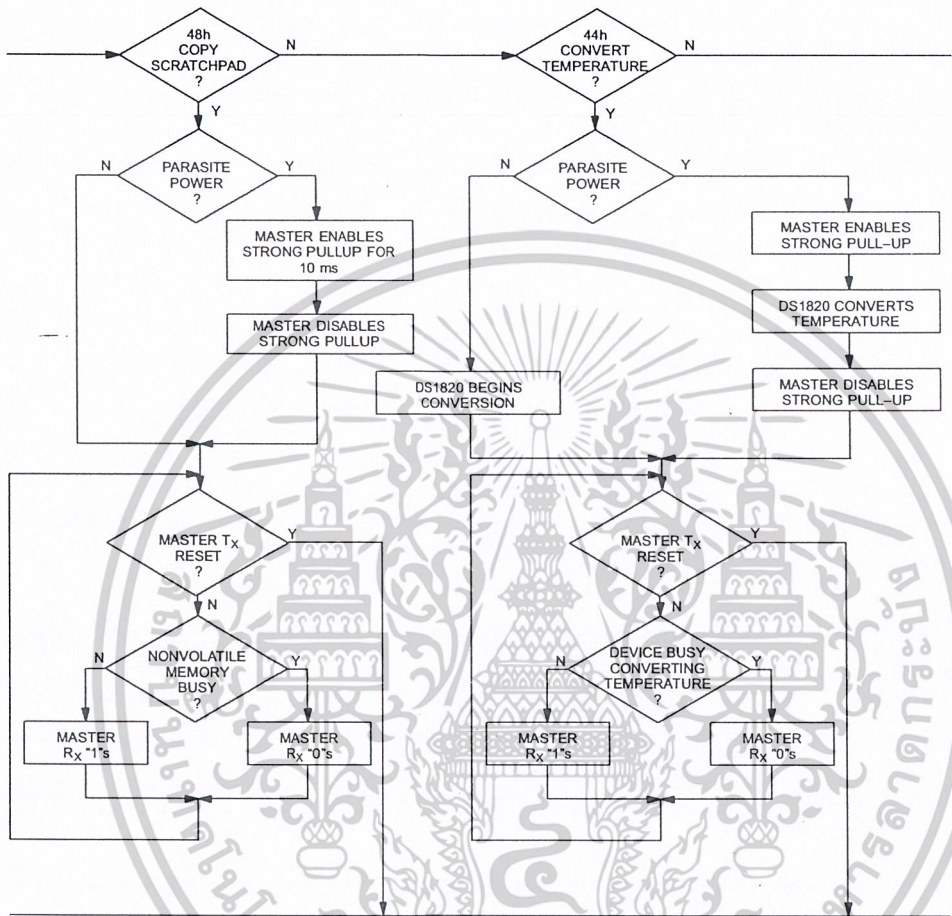
Write Scratchpad [4Eh]

This command writes to the scratchpad of the DS1820, starting at address 2. The next two bytes written will be saved in scratchpad memory, at address locations 2 and 3. Writing may be terminated at any point by issuing a reset.

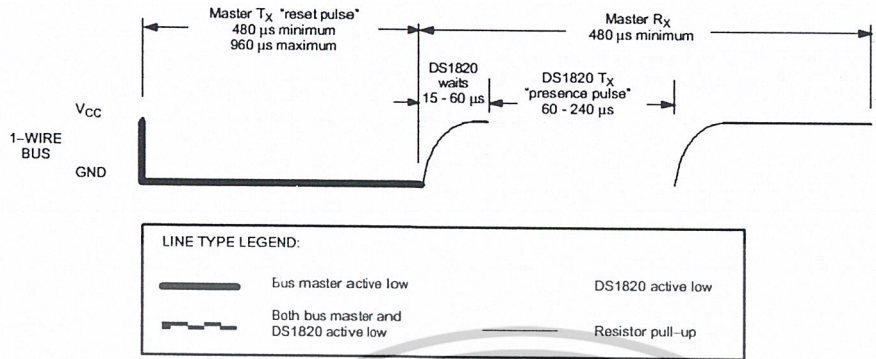




MEMORY FUNCTIONS FLOW CHART Figure 10 (cont'd)



INITIALIZATION PROCEDURE “RESET AND PRESENCE PULSES” Figure 11



DS1820 COMMAND SET Table 2

INSTRUCTION	DESCRIPTION	PROTOCOL	1-WIRE BUS AFTER ISSUING PROTOCOL	NOTES
TEMPERATURE CONVERSION COMMANDS				
Convert T	Initiates temperature conversion.	44h	<read temperature busy status>	1
MEMORY COMMANDS				
Read Scratchpad	Reads bytes from scratchpad and reads CRC byte.	BEh	<read data up to 9 bytes>	
Write Scratchpad	Writes bytes into scratchpad at addresses 2 and 3 (TH and TL temperature triggers).	4Eh	<write data into 2 bytes at addr. 2 and addr. 3>	
Copy Scratchpad	Copies scratchpad into nonvolatile memory (addresses 2 and 3 only).	48h	<read copy status>	2
Recall E ²	Recalls values stored in nonvolatile memory into scratchpad (temperature triggers).	B8h	<read temperature busy status>	
Read Power Supply	Signals the mode of DS1820 power supply to the master.	B4h	<read supply status>	

- NOTES:
1. Temperature conversion takes up to 500 ms. After receiving the Convert T protocol, if the part does not receive power from the V_{DD} pin, the I/O line for the DS1820 must be held high for at least 500 ms to provide power during the conversion process. As such, no other activity may take place on the 1-Wire bus for at least this period after a Convert T command has been issued.
 2. After receiving the Copy Scratchpad protocol, if the part does not receive power from the V_{DD} pin, the I/O line for the DS1820 must be held high for at least 10 ms to provide power during the copy process. As such, no other activity may take place on the 1-Wire bus for at least this period after a Copy Scratchpad command has been issued.

Read Scratchpad [BEh]

This command reads the contents of the scratchpad. Reading will commence at byte 0, and will continue through the scratchpad until the 9th (byte=8, CRC) byte is read. If not all locations are to be read, the master may issue a reset to terminate reading at any time.

Copy Scratchpad [48h]

This command copies the scratchpad into the E² memory of the DS1820, storing the temperature trigger bytes in nonvolatile memory. If the bus master issues read time slots following this command, the DS1820 will output "0" on the bus as long as it is busy copying the scratchpad to E²; it will return a "1" when the copy process is complete. If parasite powered, the bus master has to enable a strong pull-up for at least 10 ms immediately after issuing this command.

Convert T [44h]

This command begins a temperature conversion. No further data is required. The temperature conversion will be performed and then the DS1820 will remain idle. If the bus master issues read time slots following this command, the DS1820 will output "0" on the bus as long as it is busy making a temperature conversion; it will return a "1" when the temperature conversion is complete. If parasite powered, the bus master has to enable a strong pullup for 500 ms immediately after issuing this command.

Recall E2 [B8h]

This command recalls the temperature trigger values stored in E² to the scratchpad. This recall operation happens automatically upon power-up to the DS1820 as well, so valid data is available in the scratchpad as soon as the device has power applied. With every read data time slot issued after this command has been sent, the device will output its temperature converter busy flag "0"=busy, "1"=ready.

Read Power Supply [B4h]

With every read data time slot issued after this command has been sent to the DS1820, the device will signal its power mode: "0"=parasite power, "1"=external power supply provided.

READ/WRITE TIME SLOTS

DS1820 data is read and written through the use of time slots to manipulate bits and a command word to specify the transaction.

Write Time Slots

A write time slot is initiated when the host pulls the data line from a high logic level to a low logic level. There are two types of write time slots: Write One time slots and Write Zero time slots. All write time slots must be a minimum of 60 μ s in duration with a minimum of a one μ s recovery time between individual write cycles.

The DS1820 samples the I/O line in a window of 15 μ s to 60 μ s after the I/O line falls. If the line is high, a Write One occurs. If the line is low, a Write Zero occurs (see Figure 12).

For the host to generate a Write One time slot, the data line must be pulled to a logic low level and then released, allowing the data line to pull up to a high level within 15 μ s after the start of the write time slot.

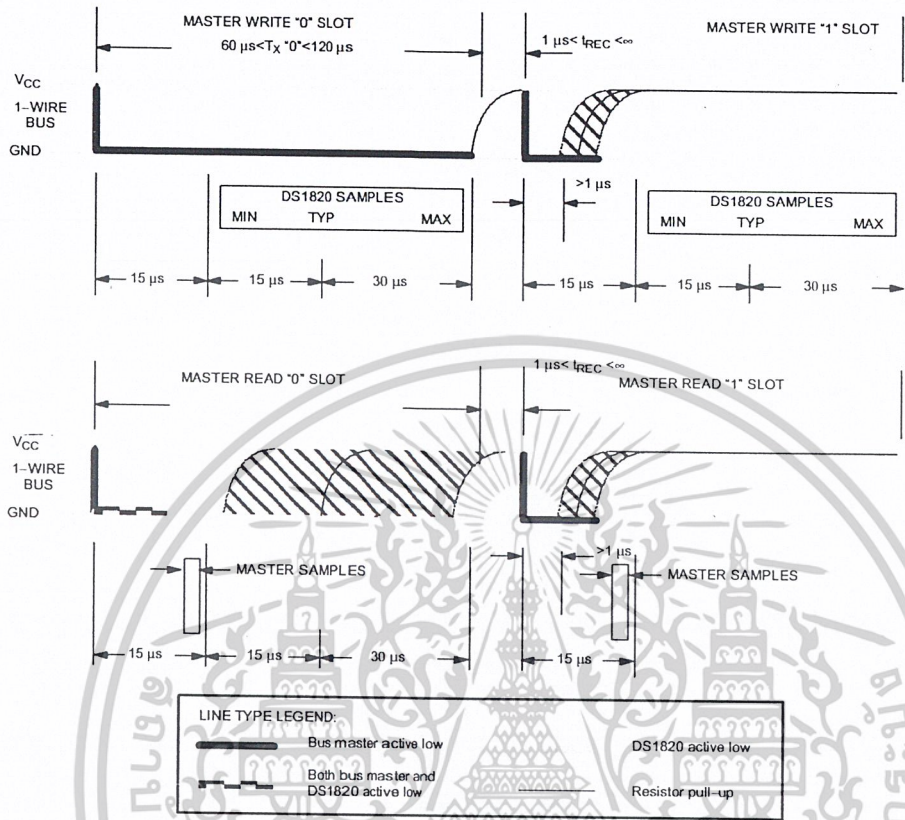
For the host to generate a Write Zero time slot, the data line must be pulled to a logic low level and remain low for 60 μ s.

Read Time Slots

The host generates read time slots when data is to be read from the DS1820. A read time slot is initiated when the host pulls the data line from a logic high level to logic low level. The data line must remain at a low logic level for a minimum of one μ s; output data from the DS1820 is valid for 15 μ s after the falling edge of the read time slot. The host therefore must stop driving the I/O pin low in order to read its state 15 μ s from the start of the read slot (see Figure 12). By the end of the read time slot, the I/O pin will pull back high via the external pull-up resistor. All read time slots must be a minimum of 60 μ s in duration with a minimum of a one μ s recovery time between individual read slots.

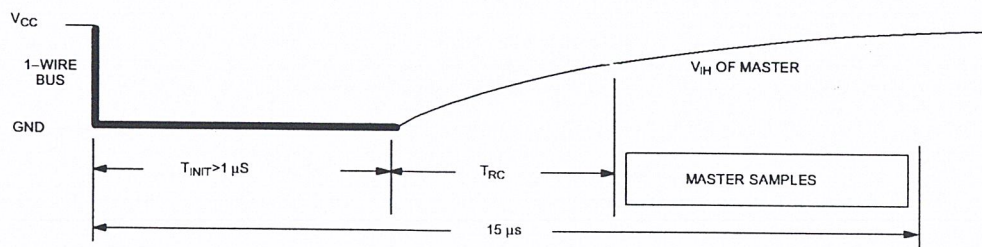
Figure 13 shows that the sum of T_{INIT} , T_{RC} , and T_{SAMPLE} must be less than 15 μ s. Figure 14 shows that system timing margin is maximized by keeping T_{INIT} and T_{RC} as small as possible and by locating the master sample time towards the end of the 15 μ s period.

READ/WRITE TIMING DIAGRAM Figure 12

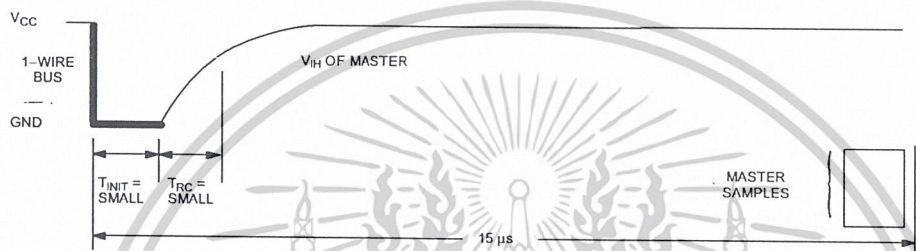


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DETAILED MASTER READ “1” TIMING Figure 13



RECOMMENDED MASTER READ “1” TIMING Figure 14



LINE TYPE LEGEND:	
	Bus master active low
	Both bus master and DS1820 active low
	DS1820 active low
	Resistor pull-up

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Related Application Notes

The following Application Notes can be applied to the DS1820. These notes can be obtained from the Dallas

Semiconductor "Application Note Book", via our website at <http://www.dalsemi.com/>, or through our faxback service at (214) 450-0441.

Application Note 27: "Understanding and Using Cyclic Redundancy Checks with Dallas Semiconductor Touch Memory Product"

Application Note 55: "Extending the Contact Range of Touch Memories"

Application Note 74: "Reading and Writing Touch Memories via Serial Interfaces"

Application Note 104: "Minimalist Temperature Control Demo"

Application Note 105: "High Resolution Temperature Measurement with Dallas Direct-to-Direct Temperature Sensors"

Application Note 106: "Complex MicroLANs"

Application Note 108: "MicroLAN – In the Long Run"

Sample 1-Wire subroutines that can be used in conjunction with AN74 can be downloaded from the website or our Anonymous FTP Site.



MEMORY FUNCTION EXAMPLE Table 3
Example: Bus Master initiates temperature conversion, then reads temperature (parasite power assumed).

MASTER MODE	DATA (LSB FIRST)	COMMENTS
TX	Reset	Reset pulse (480–960 μ s).
RX	Presence	Presence pulse.
TX	55h	Issue "Match ROM" command.
TX	<64-bit ROM code>	Issue address for DS1820.
TX	44h	Issue "Convert T" command.
TX	<I/O LINE HIGH>	I/O line is held high for at least 500 ms by bus master to allow conversion to complete.
TX	Reset	Reset pulse.
RX	Presence	Presence pulse.
TX	55h	Issue "Match ROM" command.
TX	<64-bit ROM code>	Issue address for DS1820.
TX	BEh	Issue "Read Scratchpad" command.
RX	<9 data bytes>	Read entire scratchpad plus CRC; the master now recalculates the CRC of the eight data bytes received from the scratchpad, compares the CRC calculated and the CRC read. If they match, the master continues; if not, this read operation is repeated.
TX	Reset	Reset Pulse.
RX	Presence	Presence pulse, done.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MEMORY FUNCTION EXAMPLE Table 4
Example: Bus Master writes memory (parasite power and only one DS1820 assumed).

MASTER MODE	DATA (LSB FIRST)	COMMENTS
TX	Reset	Reset pulse.
RX	Presence	Presence pulse.
TX	CCh	Skip ROM command.
TX	4Eh	Write Scratchpad command.
TX	<2 data bytes>	Writes two bytes to scratchpad (TH and TL).
TX	Reset	Reset pulse.
RX	Presence	Presence pulse.
TX	CCh	Skip ROM command.
TX	BEh	Read Scratchpad command.
RX	<9 data bytes>	Read entire scratchpad plus CRC. The master now recalculates the CRC of the eight data bytes received from the scratchpad, compares the CRC and the two other bytes read back from the scratchpad. If data match, the master continues; if not, repeat the sequence.
TX	Reset	Reset pulse.
RX	Presence	Presence pulse.
TX	CCh	Skip ROM command.
TX	48h	Copy Scratchpad command; after issuing this command, the master must wait 6 ms for copy operation to complete.
TX	Reset	Reset pulse.
RX	Presence	Presence pulse, done.

MEMORY FUNCTION EXAMPLE Table 5
Example: Temperature conversion and interpolation (external power supply and only one DS1820 assumed).

MASTER MODE	DATA (LSB FIRST)	COMMENTS
TX	Reset	Reset pulse.
TR	Presence	Presence pulse.
TX	CCh	Skip ROM command.
TX	44h	Convert T command.
RX	<1 data byte>	Read busy flag eight times. The master continues reading one byte (or bit) after another until the data is FFh (all bits 1).
TX	Reset	Reset pulse.
RX	Presence	Presence pulse.
TX	CCh	Skip ROM command.
TX	BEh	Read Scratchpad command.
RX	<9 data bytes>	Read entire scratchpad plus CRC. The master now recalculates the CRC of the eight data bytes received from the scratchpad and compares both CRCs. If the CRCs match, the data is valid. The master saves the temperature value and stores the contents of the count register and count per °C register as COUNT_REMAIN and COUNT_PER_C, respectively.
TX	Reset	Reset pulse.
RX	Presence	Presence pulse, done.
—	—	CPU calculates temperature as described in the data sheet for higher resolution.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	−0.5V to +7.0V
Operating Temperature	−55°C to +125°C
Storage Temperature	−55°C to +125°C
Soldering Temperature	260°C for 10 seconds

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

RECOMMENDED DC OPERATING CONDITIONS

PARAMETER	SYMBOL	CONDITION	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V _{DD}	I/O Functions	2.8	5.0	5.5	V	1, 2
		±1/2°C Accurate Temperature Conversions	4.3		5.5		
Data Pin	I/O		−0.5		+5.5	V	2
Logic 1	V _{IH}		2.0		V _{CC} +0.3	V	2, 3
Logic 0	V _{IL}		−0.3		+0.8	V	2, 4

DC ELECTRICAL CHARACTERISTICS (−55°C to +125°C; V_{DD}=3.6V to 5.5V)

PARAMETER	SYMBOL	CONDITION	MIN	TYP	MAX	UNITS	NOTES
Thermometer Error	I _{ERR}	−0°C to +70°C			±1/2	°C	1, 9, 10
		−55°C to 0°C and +70°C to +125°C			See Typical Curve		
Input Logic High	V _{IH}		2.2		5.5	V	2, 3
Input Logic Low	V _{IL}		−0.3		+0.8	V	2, 4
Sink Current	I _L	V _{I/O} =0.4V	−4.0			mA	2
Standby Current	I _Q			200	350	nA	8
Active Current	I _{DD}			1	1.5	mA	5, 6
Input Load Current	I _L			5		μA	7

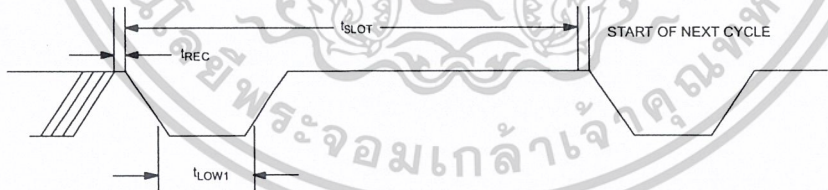
AC ELECTRICAL CHARACTERISTICS: (–55°C to +125°C; V_{DD}=3.6V to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Temperature Conversion Time	t _{CONV}		200	500	ms	
Time Slot	t _{SLOT}	60		120	μs	
Recovery Time	t _{REC}	1			μs	
Write 0 Low Time	t _{LOW0}	60		120	μs	
Write 1 Low Time	t _{LOW1}	1		15	μs	
Read Data Valid	t _{RDV}			15	μs	
Reset Time High	t _{RSTH}	480			μs	
Reset Time Low	t _{RSTL}	480		4800	μs	
Presence Detect High	t _{PDHIGH}	15		60	μs	
Presence Detect Low	t _{PDLOW}	60		240	μs	
Capacitance	C _{IN/OUT}			25	pF	

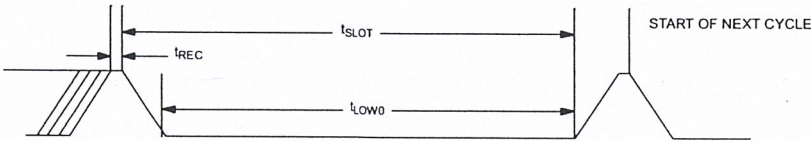
NOTES:

1. Temperature conversion will work with ±2°C accuracy down to V_{DD} = 3.4 volts.
2. All voltages are referenced to ground.
3. Logic one voltages are specified at a source current of 1 mA.
4. Logic zero voltages are specified at a sink current of 4 mA.
5. I_{DD} specified with V_{CC} at 5.0 volts.
6. Active current refers to either temperature conversion or writing to the E² memory. Writing to E² memory consumes approximately 200 μA for up to 10 ms.
7. Input load is to ground.
8. Standby current specified up to 70°C. Standby current typically is 5 μA at 125°C.
9. See Typical Curve for specification limits outside the 0°C to 70°C range. Thermometer error reflects sensor accuracy as tested during calibration.
10. Typical accuracy curve valid for 4.3V ≤ V_{DD} ≤ 5.5V.

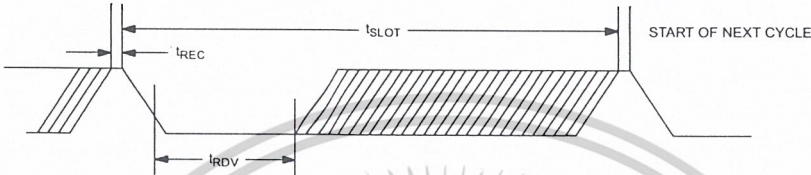
1-WIRE WRITE ONE TIME SLOT



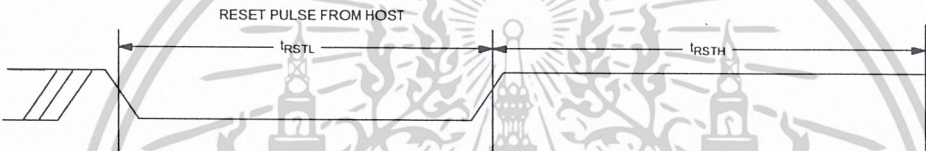
1-WIRE WRITE ZERO TIME SLOT



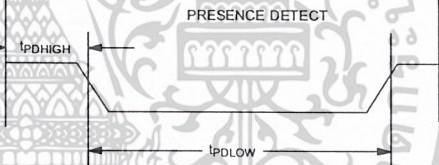
1-WIRE READ ZERO TIME SLOT



1-WIRE RESET PULSE

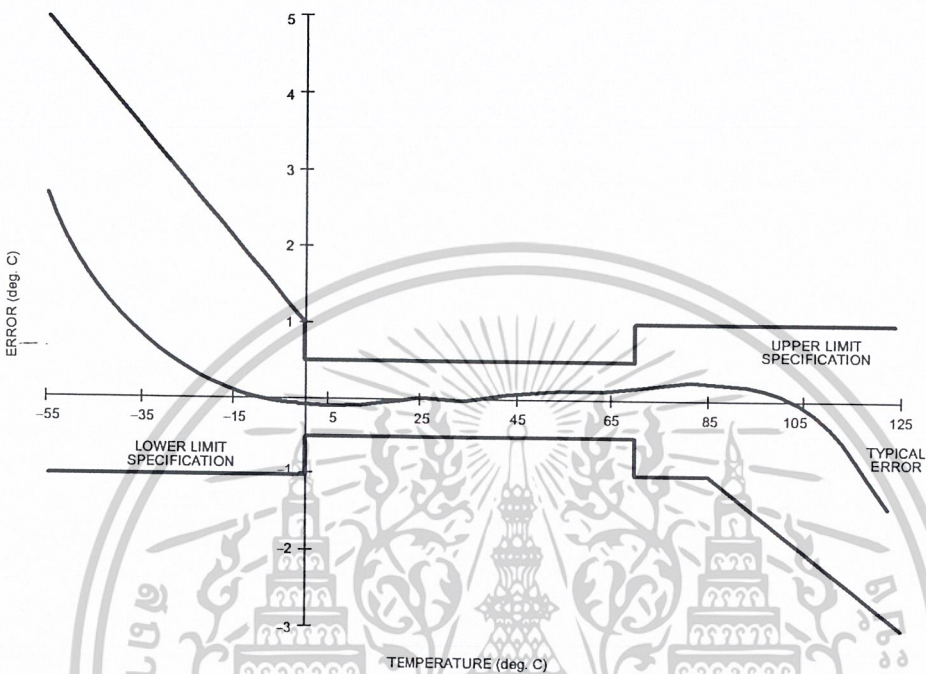


1-WIRE PRESENCE DETECT



TYPICAL PERFORMANCE CURVE

DS1820 DIGITAL THERMOMETER AND THERMOSTAT
TEMPERATURE READING ERROR



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



6-Pin DIP Optoisolators Transistor Output

The 4N25/A, 4N26, 4N27 and 4N28 devices consist of a gallium arsenide infrared emitting diode optically coupled to a monolithic silicon phototransistor detector.

- Most Economical Optoisolator Choice for Medium Speed, Switching Applications
- Meets or Exceeds All JEDEC Registered Specifications
- *To order devices that are tested and marked per VDE 0884 requirements, the suffix "V" must be included at end of part number. VDE 0884 is a test option.*

Applications

- General Purpose Switching Circuits
- Interfacing and coupling systems of different potentials and impedances
- I/O Interfacing
- Solid State Relays

MAXIMUM RATINGS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Rating	Symbol	Value	Unit
--------	--------	-------	------

INPUT LED

Reverse Voltage	V_R	3	Volts
Forward Current — Continuous	I_F	60	mA
LED Power Dissipation @ $T_A = 25^\circ\text{C}$ with Negligible Power in Output Detector Derate above 25°C	P_D	120	mW
		1.41	mW/ $^\circ\text{C}$

OUTPUT TRANSISTOR

Collector–Emitter Voltage	V_{CEO}	30	Volts
Emitter–Collector Voltage	V_{ECO}	7	Volts
Collector–Base Voltage	V_{CBO}	70	Volts
Collector Current — Continuous	I_C	150	mA
Detector Power Dissipation @ $T_A = 25^\circ\text{C}$ with Negligible Power in Input LED Derate above 25°C	P_D	150	mW
		1.76	mW/ $^\circ\text{C}$

TOTAL DEVICE

Isolation Surge Voltage ⁽¹⁾ (Peak ac Voltage, 60 Hz, 1 sec Duration)	V_{ISO}	7500	Vac(pk)
Total Device Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	250 2.94	mW mW/ $^\circ\text{C}$
Ambient Operating Temperature Range ⁽²⁾	T_A	–55 to +100	$^\circ\text{C}$
Storage Temperature Range ⁽²⁾	T_{stg}	–55 to +150	$^\circ\text{C}$
Soldering Temperature (10 sec, 1/16" from case)	T_L	260	$^\circ\text{C}$

1. Isolation surge voltage is an internal device dielectric breakdown rating.
For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.
2. Refer to Quality and Reliability Section in Opto Data Book for information on test conditions.

Preferred devices are Motorola recommended choices for future use and best overall value.
GlobalOptoisolator is a trademark of Motorola, Inc.

REV 5

© Motorola, Inc. 1995

4N25*

4N25A*

4N26*

[CTR = 20% Min]

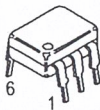
4N27

4N28

[CTR = 10% Min]

*Motorola Preferred Devices

STYLE 1 PLASTIC



STANDARD THRU HOLE
CASE 730A–04

SCHEMATIC



- PIN 1. LED ANODE
2. LED CATHODE
3. N.C.
4. EMITTER
5. COLLECTOR
6. BASE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



MOTOROLA

4N25 4N25A 4N26 4N27 4N28

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)(1)

Characteristic	Symbol	Min	Typ(1)	Max	Unit
INPUT LED					
Forward Voltage (I _F = 10 mA)	V _F	—	1.15	1.5	Volts
T _A = 25°C		—	1.3	—	
T _A = -55°C		—	1.05	—	
T _A = 100°C					
Reverse Leakage Current (V _R = 3 V)	I _R	—	—	100	μA
Capacitance (V = 0 V, f = 1 MHz)	C _J	—	18	—	pF
OUTPUT TRANSISTOR					
Collector–Emitter Dark Current (V _{CE} = 10 V, T _A = 25°C)	I _{CEO}	—	1	50	nA
4N25,25A,26,27		—	1	100	
4N28					
(V _{CE} = 10 V, T _A = 100°C)	I _{CEO}	—	1	—	μA
Collector–Base Dark Current (V _{CB} = 10 V)	I _{CBO}	—	0.2	—	nA
Collector–Emitter Breakdown Voltage (I _C = 1 mA)	V _{(BR)CEO}	30	45	—	Volts
Collector–Base Breakdown Voltage (I _C = 100 μA)	V _{(BR)CBO}	70	100	—	Volts
Emitter–Collector Breakdown Voltage (I _E = 100 μA)	V _{(BR)ECO}	7	7.8	—	Volts
DC Current Gain (I _C = 2 mA, V _{CE} = 5 V)	h _{FE}	—	500	—	—
Collector–Emitter Capacitance (f = 1 MHz, V _{CE} = 0)	C _{CE}	—	7	—	pF
Collector–Base Capacitance (f = 1 MHz, V _{CB} = 0)	C _{CB}	—	19	—	pF
Emitter–Base Capacitance (f = 1 MHz, V _{EB} = 0)	C _{EB}	—	9	—	pF
COUPLED					
Output Collector Current (I _F = 10 mA, V _{CE} = 10 V)	I _C (CTR)(2)	2 (20)	7 (70)	—	mA (%)
4N25,25A,26		1 (10)	5 (50)	—	
4N27,28					
Collector–Emitter Saturation Voltage (I _C = 2 mA, I _F = 50 mA)	V _{CE(sat)}	—	0.15	0.5	Volts
Turn–On Time (I _F = 10 mA, V _{CC} = 10 V, R _L = 100 Ω)(3)	t _{on}	—	2.8	—	μs
Turn–Off Time (I _F = 10 mA, V _{CC} = 10 V, R _L = 100 Ω)(3)	t _{off}	—	4.5	—	μs
Rise Time (I _F = 10 mA, V _{CC} = 10 V, R _L = 100 Ω)(3)	t _r	—	1.2	—	μs
Fall Time (I _F = 10 mA, V _{CC} = 10 V, R _L = 100 Ω)(3)	t _f	—	1.3	—	μs
Isolation Voltage (f = 60 Hz, t = 1 sec)(4)	V _{ISO}	7500	—	—	Vac(pk)
Isolation Resistance (V = 500 V)(4)	R _{ISO}	10 ¹¹	—	—	Ω
Isolation Capacitance (V = 0 V, f = 1 MHz)(4)	C _{ISO}	—	0.2	—	pF

1. Always design to the specified minimum/maximum electrical limits (where applicable).
2. Current Transfer Ratio (CTR) = I_C/I_F × 100%.
3. For test circuit setup and waveforms, refer to Figure 11.
4. For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL CHARACTERISTICS

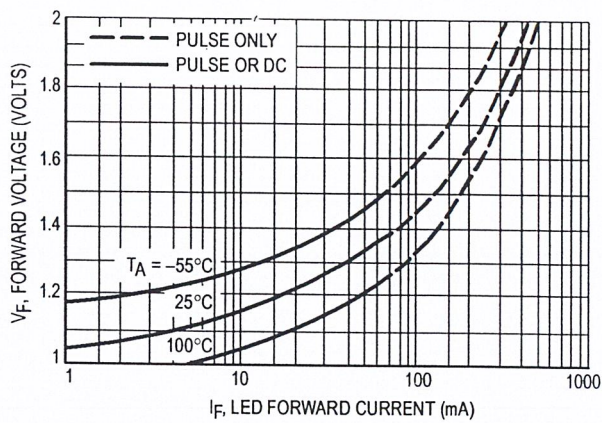


Figure 1. LED Forward Voltage versus Forward Current

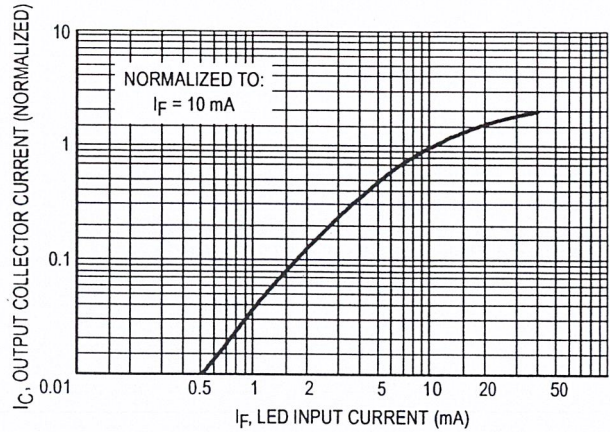


Figure 2. Output Current versus Input Current

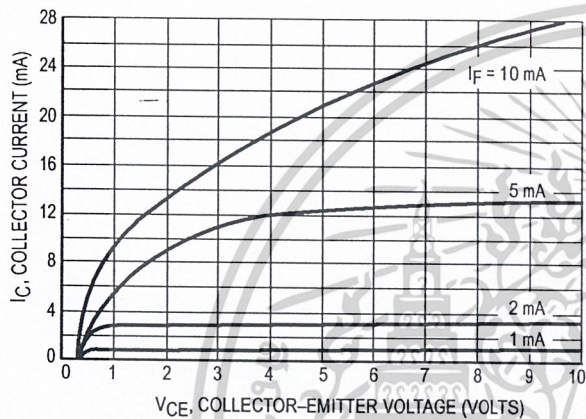


Figure 3. Collector Current versus Collector-Emitter Voltage

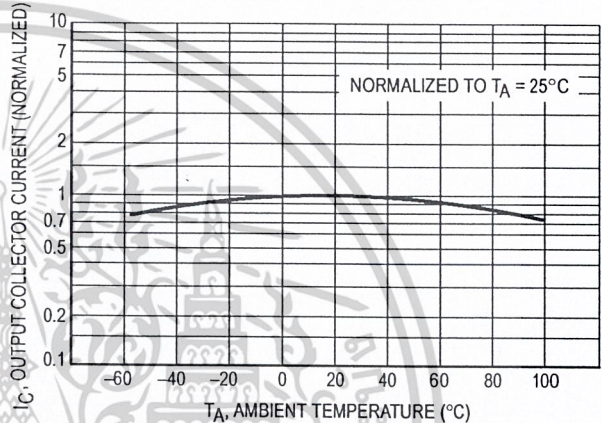


Figure 4. Output Current versus Ambient Temperature

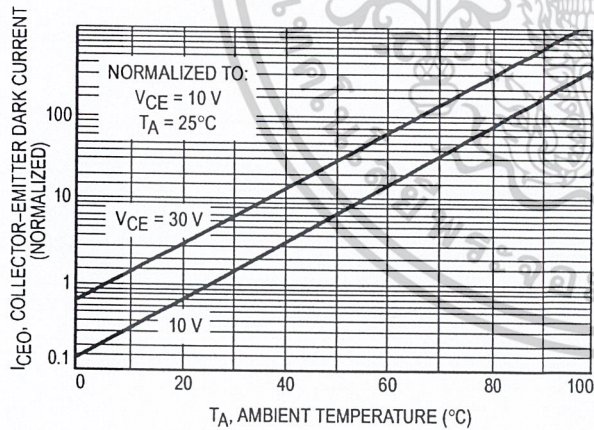


Figure 5. Dark Current versus Ambient Temperature

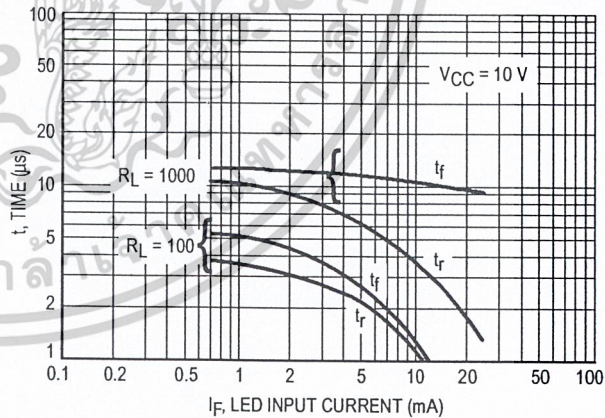


Figure 6. Rise and Fall Times (Typical Values)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4N25 4N25A 4N26 4N27 4N28

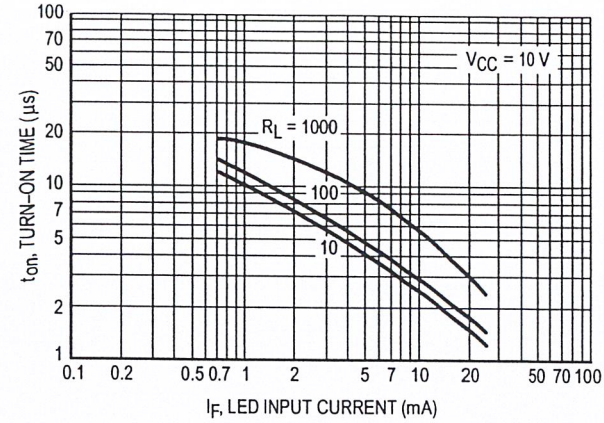


Figure 7. Turn-On Switching Times (Typical Values)

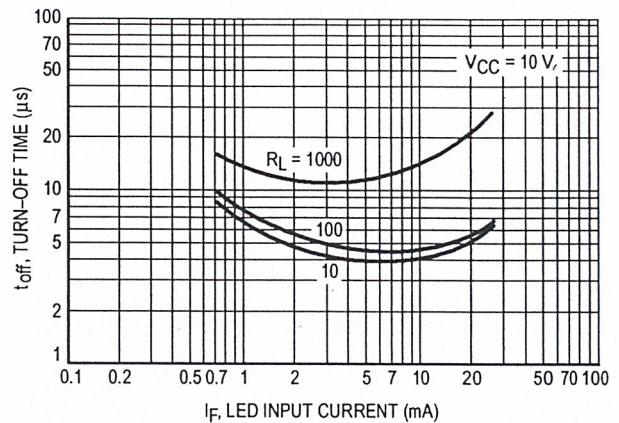


Figure 8. Turn-Off Switching Times (Typical Values)

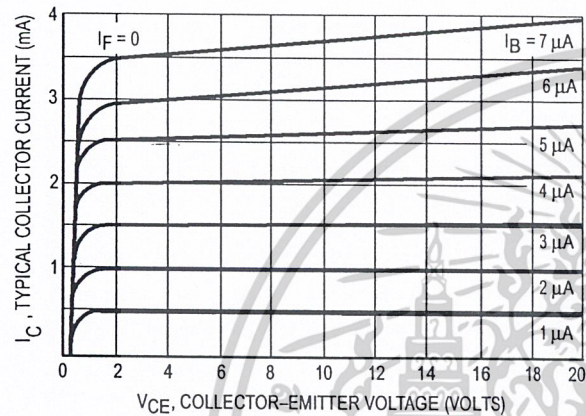


Figure 9. DC Current Gain (Detector Only)

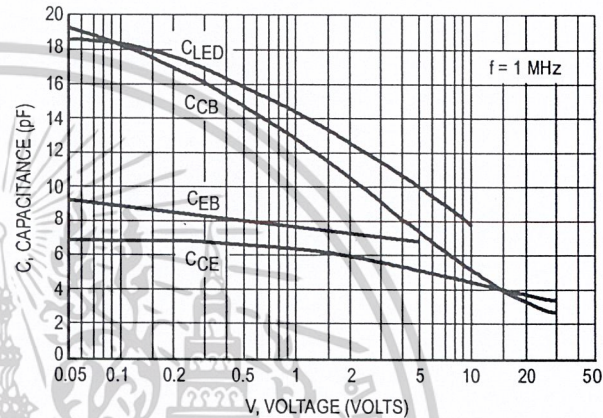


Figure 10. Capacitances versus Voltage

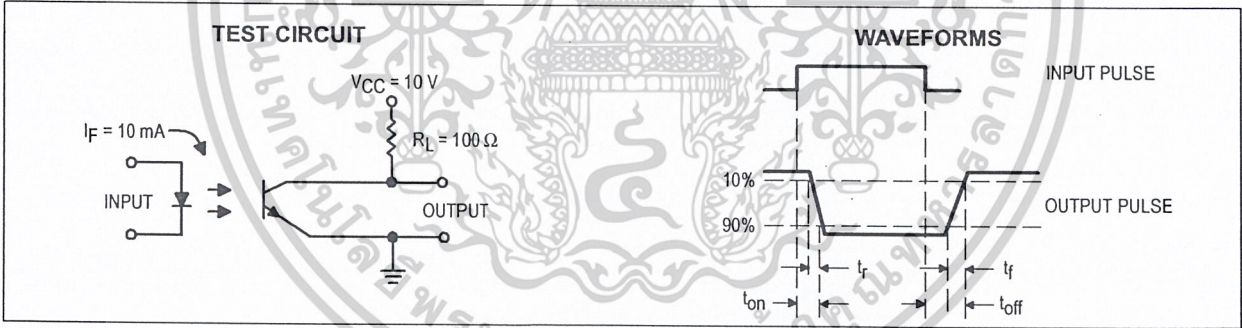
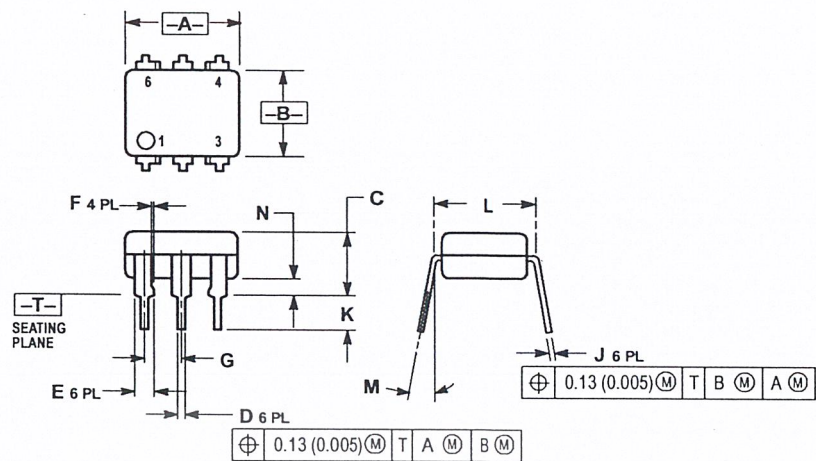


Figure 11. Switching Time Test Circuit and Waveforms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PACKAGE DIMENSIONS

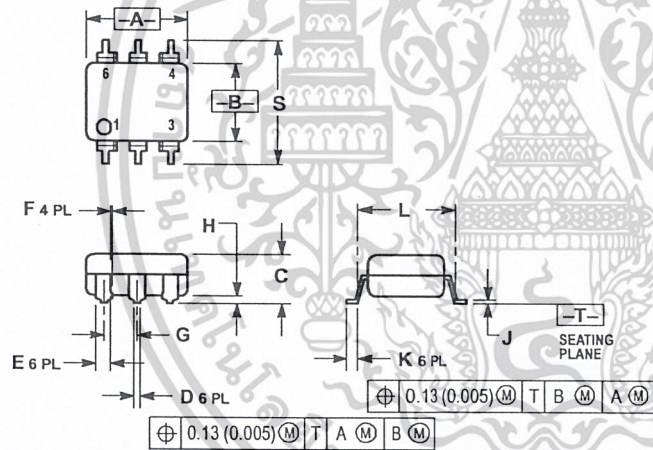


- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
 2. CONTROLLING DIMENSION: INCH.
 3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.

DIM	MIN	MAX	MIN	MAX
A	0.320	0.350	8.13	8.89
B	0.240	0.260	6.10	6.60
C	0.115	0.200	2.93	5.08
D	0.016	0.020	0.41	0.50
E	0.040	0.070	1.02	1.77
F	0.010	0.014	0.25	0.36
G	0.100 BSC		2.54 BSC	
J	0.008	0.012	0.21	0.30
K	0.100	0.150	2.54	3.81
L	0.300 BSC		7.62 BSC	
M	0°	15°	0°	15°
N	0.015	0.100	0.38	2.54

- STYLE 1:
- PIN 1. ANODE
 - CATHODE
 - NC
 - EMITTER
 - COLLECTOR
 - BASE

CASE 730A-04
ISSUE G



- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
 2. CONTROLLING DIMENSION: INCH.

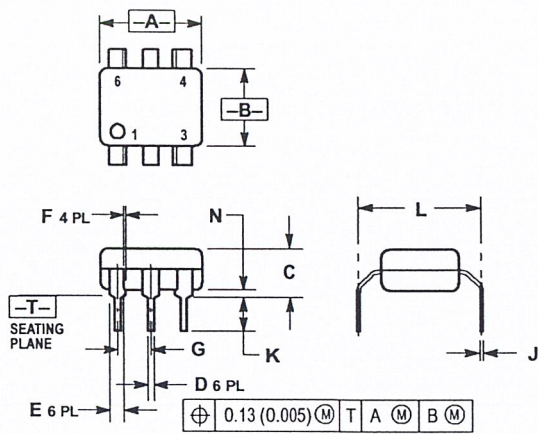
DIM	MIN	MAX	MIN	MAX
A	0.320	0.350	8.13	8.89
B	0.240	0.260	6.10	6.60
C	0.115	0.200	2.93	5.08
D	0.016	0.020	0.41	0.50
E	0.040	0.070	1.02	1.77
F	0.010	0.014	0.25	0.36
G	0.100 BSC		2.54 BSC	
H	0.020	0.025	0.51	0.63
J	0.008	0.012	0.20	0.30
K	0.008	0.035	0.16	0.88
L	0.320 BSC		8.13 BSC	
S	0.332	0.390	8.43	9.90

*Consult factory for leadform option availability

CASE 730C-04
ISSUE D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4N25 4N25A 4N26 4N27 4N28



- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
 2. CONTROLLING DIMENSION: INCH.
 3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.320	0.350	8.13	8.89
B	0.240	0.260	6.10	6.60
C	0.115	0.200	2.93	5.08
D	0.016	0.020	0.41	0.50
E	0.040	0.070	1.02	1.77
F	0.010	0.014	0.25	0.36
G	0.100 BSC		2.54 BSC	
J	0.008	0.012	0.21	0.30
K	0.100	0.150	2.54	3.81
L	0.400	0.425	10.16	10.80
N	0.015	0.040	0.38	1.02

*Consult factory for leadform option availability

CASE 730D-05
ISSUE D

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and  are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

How to reach us:
USA/EUROPE: Motorola Literature Distribution;
P.O. Box 20912; Phoenix, Arizona 85036. 1-800-441-2447

JAPAN: Nippon Motorola Ltd.; Tatsumi-SPD-JLDC, Toshikatsu Otsuki,
6F Seibu-Butsuryu-Center, 3-14-2 Tatsumi Koto-Ku, Tokyo 135, Japan. 03-3521-8315

MFAX: RMFAX0@email.sps.mot.com - TOUCHTONE (602) 244-6609
INTERNET: <http://Design-NET.com>

HONG KONG: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park,
51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852-26629298



4N25/D



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

#include <NEW14.H>
#include <LCD.C>

int a,z,x,q,w,e,t,temp,index,check,value,value_col,row0,value_row,row,col;
int h1,h2,h3,d,ack,noack,error,key_press,put,i,send;
int h,o,u,j,air;
int tempdi,key_humid;
long tem,temped,unoo,key_temp,wanted,key;
float tem1,tem2,t_c,rh_lin;
short dq,gotemp,gohumid;

#byte port_c=0x07
#bit wire_ds = 9.0
#bit wire_coil = 8.0
#bit wire_humid = 8.1
#bit wire_fan = 8.2
#bit wire_fann = 8.3
#bit data=8.4
#bit sck=8.5
#bit wire_ds = 9.0

void s_transstart(void){
    set_tris_d(0x00);
    data=1;
    sck=0;
    delay_us(1);
    sck=1;
    delay_us(1);
    data=0;
    delay_us(1);
    sck=0;
    delay_us(6);
    sck=1;
    delay_us(1);
    data=1;
    delay_us(1);
    sck=0;
}

void s_write_byte(int value){
    int i,val;
    i,val=0;
    set_tris_d(0x00);
    for(i=0x80;i>0;i/=2){
        if(i&value)data=1;
        else data=0;
        sck=1;
        delay_us(6);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        sck=0;
    }
    data=1;
    sck=1;
    sck=0;
}
char s_read_byte(char ack){
char i,val;
short y;
    set_tris_d(0x00);
    data=1;
    val = 0;
    for(i=0x80;i>0;i/=2){
        sck=1;
        set_tris_d(0x10);
        y=data;
        if(y)
            val=(val|i);
        sck=0;
    }
    data=!ack;
    set_tris_d(0x00);
    sck=1;
    delay_us(6);
    sck=0;
    data=1;
    return val;
}
void s_connectionreset(void){
char i;
    set_tris_d(0x00);
    data=1;
    sck=0;
    for(i=0;i<9;i++){
        sck=1;
        sck=0;
    }
    s_transstart();
}
void s_measure(void){
short y;
long i;
    error=0;
    s_transstart();
    s_write_byte(0x05);
    set_tris_d(0x10);
    for(i=0;i<65535;i++){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

    y=data;
    if(y==0)
        break;
    }
    if(y){error=1;}
    h1=s_read_byte(ack);
    h2=s_read_byte(ack);
    h3=s_read_byte(noack);
    }
void converttfloat(void){
    tem=h1;
    tem=tem<<8;
    tem=tem|h2;
    tem2=tem;
}
void calc_sht11(void){
    rh_lin=(-0.0000028)*tem2*tem2+0.0405*tem2+(-4.0);
}
void in_point(void){
    h=rh_lin/100;
    o=rh_lin-(h*100);
    u=o/10;
    j=o%10;
    h=0x30+h;
    u=0x30+u;
    j=0x30+j;
    lcd_gotoxy(11,2);
    lcd_putc(h);
    lcd_gotoxy(12,2);
    lcd_putc(u);
    lcd_gotoxy(13,2);
    lcd_putc(j);
    lcd_gotoxy(14,2);
    lcd_putc("%");
    delay_ms(1000);
}
void humidshow(void){
    delay_ms(2);
    s_connectionreset();
    s_measure();
    converttfloat();
    calc_sht11();
    in_point();
}

void ds_hi(void){
    set_tris_e(0x01);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

}
void ds_low(void){
    set_tris_e(0x00);
    wire_ds=0;
}
void ds_initial(void){
    ds_hi();
    ds_low();
    delay_us(500);
    ds_hi();
    delay_us(150);
    set_tris_e(0x01);
    dq=wire_ds;
}
void outport(int v){
    int w,i,a,b;
    b=v;
    for(i=1;i<9;i++){
        a=b&0x01;
        if(a==0x01){
            w=1;
            set_tris_e(0x00);
            wire_ds=0;
            delay_us(4);
            wire_ds=1;
            delay_us(75);
        }
        if(a==0x00){
            w=0;
            set_tris_e(0x00);
            wire_ds=0;
            delay_us(75);
            wire_ds=1;
            delay_us(4);
        }
        b=b>>1;
    }
}
void strong(void){
    set_tris_e(0x00);
    wire_ds=1;
    delay_ms(750);
}
void read(void){
    long z;
    int i,g;
    z=0x0000;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

for(i=1;i<17;i++){
    set_tris_e(0x00);
    wire_ds=0;
    delay_us(2);
    wire_ds=1;
    delay_us(4);
    set_tris_e(0x01);
    g=wire_ds;
    z=z>>1;
    if(g==0x01)
        z=z|0x8000;
    if(g==0x00)
        z=z|0x0000;
    delay_us(45);
}
tempdi=z;
}
void showbit(int b){
int w,i,a,q,p,o,h,d,k,c,l;
long s;
p=1;
s=0;
o=0;
h=0;
d=b&0x01;
if(d==1){
    k=5;
    b=b>>1;
}
if(d==0){
    k=0;
    b=b>>1;
}
for(i=1;i<9;i++){
    if(i==1){
        p=1;
    }
    if(i>1){
        p=p*2;
    }
    a=b&0x01;
    if(a==0x01){
        w=1;
        s=s+p;
    }
    if(a==0x00){
        w=0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

    }
    b=b>>1;
}
temped=(s*10)+k;
unoo=temped;
c=s/100;
l=s-(c*100);
c=0x30+c;
o=l/10;
o=0x30+o;
h=s%10;
h=0x30+h;
lcd_gotoxy(11,1);
lcd_putc(c);
lcd_gotoxy(12,1);
lcd_putc(o);
lcd_gotoxy(13,1);
lcd_putc(h);
lcd_gotoxy(14,1);
lcd_putc(".");
k=0x30+k;
lcd_gotoxy(15,1);
lcd_putc(k);
}
void showtemp(void){
    ds_initial();
    if(dq==1){
        ds_initial();
    }
    if(dq==0){
        delay_us(450);
    }
    output(0xcc);
    output(0x44);
    strong();
    ds_initial();
    output(0xcc);
    output(0xbe);
    read();
    showbit(tempdi);
    ds_initial();
    delay_ms(1700);
}
void scankey(void){
    if(index==1){
        port_c=0x7e;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

if(index==2){
    port_c=0x7d;
}
if(index==3){
    port_c=0x7b;
    index=0;
}
index=index+1;
temp=port_c;
delay_ms(2);
value=port_c;
row0=temp&0x7f;
value_row=temp&0x78;
value_col=value&0x07;
if(key_press == 0){
    if(row0==0x75){
        row=11;
        key_press=1;
        send = 1;
    }
    switch(value_col){
        case 0x06 : col =1;
            break;
        case 0x05 : col =2;
            break;
        case 0x03 : col =3;
            break;
    }
    switch(value_row){
        case 0x38 : row =0+col;
            key_press=1;
            send = 1;
            break;
        case 0x58 : row =3+col;
            key_press=1;
            send = 1;
            break;
        case 0x68 : row =6+col;
            key_press=1;
            send=1;
            break;
    }
}
if(row==11){
    row=0;
}
put=0x30+row;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

}
void Humid(void){
int y,q;
q=1;
y=0;
lcd_putc("\f");
while(q==1){
    lcd_gotoxy(2,1);
    lcd_putc("Enter Humid.");
    scankey();
    wire_coil=0;
    if(row0==0x73){
        if(y==1){
            q=5;
        }
    }
}
if(key_press==1){
    if(send == 1){
        lcd_gotoxy(i,2);
        lcd_putc(put);
        send =0;
        if(i==1){
            wanted=row*10;
            lcd_gotoxy(2,2);
            lcd_putc("");
            y=0;
        }
        if(i==2){
            wanted=wanted+row;
            key_humid=wanted;
            y=1;
        }
        i=i+1;
        if(i==3){
            wanted=0;
            i=1;
        }
    }
    port_c = 0x78;
    delay_ms(2);
    check = port_c & 0x78;
    if(check == 0x78){
        key_press = 0;
    }
}
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

void Temperature(void){
int h,y;
t=0;
wanted=0;
key_temp=0;
y=0;
h=0;
lcd_putc("\f");
while(h<3){
    lcd_gotoxy(2,1);
    lcd_putc("Enter Temp.");
    scankey();
    if(row0==0x73){
        if(y==1){
            h=5;
        }
    }
    if(key_press==1){
        if(send == 1){
            lcd_gotoxy(i,2);
            lcd_putc(put);
            send =0;
            if(i==1){
                wanted=row*10;
                lcd_gotoxy(2,2);
                lcd_putc("");
                y=0;
            }
            if(i==2){
                wanted=wanted+row;
                t=wanted;
                key_temp=wanted*10;
                y=1;
            }
            i=i+1;
            if(i==3){
                wanted=0;
                i=1;
            }
        }
        port_c = 0x78;
        delay_ms(2);
        check = port_c & 0x78;
        if(check == 0x78){
            key_press = 0;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

    }
}
void compare_a(void){
int c;
long k,t,erase;
    k=key_temp;
    erase=0;
    showtemp();
    t=unoo;
    if(t<k){
        c=0;
    }
    if(t>k){
        c=1;
    }
    switch(c){
        case 1 : erase=t-k;
            delay_us(10);
            if(erase>50){
                wire_fan=1;
                wire_fann=1;
                wire_coil=0;
                delay_us(10);
                break;
            }
            delay_us(10);
            if(erase<=50){
                wire_fan=1;
                wire_fann=1;
                wire_coil=0;
                delay_ms(500);
                wire_fan=0;
                wire_fann=0;
                wire_coil=0;
                delay_ms(300);
                break;
            }
            delay_us(10);
        case 0 : erase=k-t;
            delay_us(10);
            if(erase>60){
                set_tris_d(0x00);
                wire_coil=1;
                wire_fan=0;
                wire_fann=0;
                break;
            }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        delay_us(10);
        if(erase<=60){
            set_tris_d(0x00);
            wire_coil=1;
            wire_fan=0;
            wire_fann=0;
            delay_ms(2500);
            wire_coil=0;
            wire_fan=0;
            wire_fann=0;
            delay_ms(400);
            wire_fan=0;
            wire_fann=0;
            break;
        }
        delay_us(10);
    }
}

void compare_b(void){
int era,humid_key,point,c,b,i;
era=0;
point=rh_lin;
humid_key=key_humid;
humidshow();
if(point<humid_key){
    i=1;
    era=humid_key-point;
    if(era>3){
        wire_humid=1;
        wire_fan=1;
        wire_fann=0;
        wire_coil=0;
    }
    if(era<=3){
        wire_humid=1;
        wire_fan=1;
        wire_fann=1;
        wire_coil=0;
        delay_ms(500);
        wire_humid=0;
        wire_fan=0;
        wire_coil=0;
    }
}
if(point>=humid_key){
    wire_humid=0;
    era=point-humid_key;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

if(era>2){
    if(i==1){
        wire_coil=1;
        wire_humid=1;
        wire_fan=1;
        delay_ms(3500);
    }
    i=i+1;
    wire_fann=0;
    wire_coil=0;
    wire_humid=0;
    wire_fan=0;
    c=1;
}
if(c==1 && era<=2){
    wire_coil=0;
    wire_humid=0;
    wire_fan=0;
    wire_fann=1;
    i=1;
}
if(era==0 && era<=2){
    c=0;
    i=0;
    wire_fan=0;
    wire_fann=1;
    wire_coil=0;
    wire_humid=0;
}
}
}
void compare_c(void){
int j,c;
long t,k,w;
t,k,w,j,c=0;
gohumid=1;
while(gotemp==0){
    showtemp();
    humidshow();
    t=unoo;
    k=key_temp;
    w=k-t;
    if(t>=k && j==1){
        gotemp=1;
        gohumid=0;
    }
    if(w>50){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

    wire_coil=1;
    wire_humid=0;
    wire_fan=0;
    wire_fann=0;
}
if(w<=20){
    wire_fan=0;
    wire_coil=0;
    wire_fann=0;
    wire_humid=0;
    delay_ms(500);
    wire_coil=1;
    j=1;
}
if(t>=k){
    c=t-k;
    if(c>=0){
        j=1;
    }
}
}
while(gohumid==0){
    showtemp();
    humidshow();
    t=rh_lin;
    k=key_humid;
    w=k-t;
    if(t>=k){
        gohumid=1;
        gotemp=0;
    }
    if(k>t){
        if(w>5){
            wire_humid=1;
            wire_fan=1;
            wire_fann=1;
            wire_coil=0;
        }
        if(w<=5){
            wire_humid=1;
            wire_fan=1;
            wire_fann=0;
            wire_coil=0;
        }
    }
}
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

void compare_Temp(void){
    while(1){
        compare_a();
    }
}
void compare_Humid(void){
    while(1){
        compare_b();
    }
}
void double_Compare(void){
    gotemp=0;
    gohumid=1;
    while(1){
        compare_c();
    }
}
main(){
    lcd_init();
    ack=1;
    noack=0;
    h,o,u,j,a,e,col,row,send,wanted,key_press=0;
    index,i=1;
    set_tris_c(0x78);
    set_tris_d(0x00);
    while(1){
        air=1;
        wire_fan=0;
        wire_fann=0;
        wire_coil=0;
        wire_humid=0;
        scankey();
        Temperature();
        if(key_temp>75){
            lcd_putc("\fLe Than 75");
            delay_ms(800);
        }
        if(key_temp<=75){
            Humid();
            while(key_humid>75){
                lcd_putc("\fLess Than 75");
                delay_ms(800);
                Humid();
            }
            if(key_humid<=70){
                lcd_putc("\f");
                z=t/10;
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

x=t%10;
z=0x30+z;
x=0x30+x;
lcd_gotoxy(1,1);
lcd_putc("Temp.");
lcd_gotoxy(7,1);
lcd_putc(z);
lcd_gotoxy(8,1);
lcd_putc(x);
lcd_gotoxy(1,2);
z=key_humid/10;
x=key_humid%10;
z=0x30+z;
x=0x30+x;
lcd_putc("Humid");
lcd_gotoxy(7,2);
lcd_putc(z);
lcd_gotoxy(8,2);
lcd_putc(x);
if(key_temp!=0 && key_humid==0){
    compare_Temp();
}
if(key_humid!=0 && key_temp==0){
    compare_Humid();
}
if(key_temp!=0 && key_humid!=0){
    double_Compare();
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

struct lcd_pin_map {
    boolean enable;
    boolean rs;
    boolean rw;
    boolean unused;
    int data : 4;
} lcd;

#byte lcd = 6

#define lcd_type 2
#define lcd_line_two 0x40
byte CONST LCD_INIT_STRING[4] = {0x20 | (lcd_type << 2), 0xc, 1, 6};

STRUCT lcd_pin_map const LCD_WRITE = {0,0,0,0,0};
STRUCT lcd_pin_map const LCD_READ = {0,0,0,0,15};
byte lcd_read_byte() {
    byte low,high;

    set_tris_b(LCD_READ);
    lcd.rw = 1;
    delay_cycles(1);
    lcd.enable = 1;
    delay_cycles(1);
    high = lcd.data;
    lcd.enable = 0;
    delay_cycles(1);
    lcd.enable = 1;
    delay_us(1);
    low = lcd.data;
    lcd.enable = 0;
    set_tris_b(LCD_WRITE);
    return( (high<<4) | low);
}

void lcd_send_nibble( byte n ) {
    lcd.data = n;
    delay_cycles(1);
    lcd.enable = 1;
    delay_us(2);
    lcd.enable = 0;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

    case '\b' : lcd_send_byte(0,0x10); break;
    default   : lcd_send_byte(1,c);   break;
}
}

```

```

char lcd_getc( byte x, byte y) {
    char value;

```

```

    lcd_gotoxy(x,y);
    lcd.rs=1;
    value = lcd_read_byte();
    lcd.rs=0;
    return(value);
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้