

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หมวดกลดอุณหภูมิต่ำ

Hypothermia Helmet



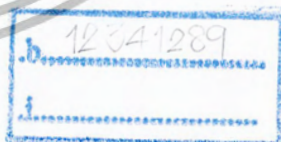
T117536



เลขหมู่.....
เลขทะเบียน.....
วัน,เดือน,ปี.....

117536

5 ต.ค. 2554



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมวกลดอุณหภูมิ

Hypothermia Helmet

นางสาวจตุพร แส่นศรี รหัส 50010189

นายจักรพงษ์ สงวนพานิช รหัส 50010203

นายฉัตรสิรินทร์ สุขมัย รหัส 50010276

นายเฉลิมพล พลอยน้ำเงิน รหัส 50010283



ปริญญาานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ ปีการศึกษา 2553

ภาควิชาอิเล็กทรอนิกส์

วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อโครงการภาษาไทย หมวดลดอุณหภูมิ

ชื่อโครงการภาษาอังกฤษ HYPOTHERMIA HELMET

ผู้จัดทำ

1. นางสาวจตุพร แสนศรี รหัส 50010189

2. นายจักรพงษ์ สงวนพานิช รหัส 50010203

3. นายจักรสิทธิ์ สุขมัย รหัส 50010276

4. นายเฉลิมพล พลอยน้ำเงิน รหัส 50010283



อาจารย์ที่ปรึกษา

(ผศ.ดร. กิติพล ชิตสุกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมวดลดอุณหภูมิ

นางสาวจตุพร	แสนศรี	50010189
นายจักรพงษ์	สงวนพานิช	50010203
นายฉัตรสิรินทร์	สุขมัย	50010276
นายเฉลิมพล	พลอยน้ำเงิน	50010283
ผศ.ดร.กิตติพล	ชิตสกุล	อาจารย์ที่ปรึกษา

ปีการศึกษา 2553

บทคัดย่อ

หมวดลดใช้เป็นอุปกรณ์ที่ได้พัฒนาขึ้นมาใช้เพื่อช่วยลดอุณหภูมิบริเวณศีรษะโดยเฉพาะเพื่อหลีกเลี่ยงการใช้น้ำ หลีกเลี่ยงการใช้ผลิตภัณฑ์ที่ใส่คล้ายกับการระบายความร้อนในเครื่องยนต์ โดยใช้ น้ำอุณหภูมิ ต่ำเป็นตัวพาความร้อนออกมา การออกแบบในเบื้องต้น ใช้ท่อพลาสติกขนาด 3 มม.ต่อเป็น โครงข่ายร่วมกับข้อต่อทำจากพลาสติก สามารถรอบพอดคล้องกับศีรษะ มีปั้มน้ำจากที่เก็บให้ไหลเวียน ภายในท่อ สามารถปรับอัตราการไหลของน้ำได้โดยการควบคุมความเร็วของมอเตอร์ มีเซ็นเซอร์วัด อุณหภูมิทั่วศีรษะ ในขั้นต้นจะ ได้ทำการทดสอบกับหัวของหุ่นจำลอง

Hypothermia Helmet

Miss Jatuporn Sansri 50010189

Mr. Jakhapong Sanguanpanich 50010203

Mr. Chutsirin Sukmai 50010276

Mr. Chalernpol Ploynumngern 50010283

Asst.Prof.Dr. Kitipol Chitsakul Advisor

Education Years 2009

ABSTRACT

Hypothermia hat is a device specially designed to reduce temperature around head for avoiding drug usage. Based on principle used for heat reduction in engine, heat transfer occurs by cooled water's circulation in plastic tubes connected as network around the head. Cold water is pumped from a reservoir with variable flow rate into the tube of 3 mm in diameter. The number of temperature sensors will be also attached around the head to verify the setting temperature. The performance of the system will be verified with human head model.

กิตติกรรมประกาศ

โครงการเรื่อง หมวกลดอุณหภูมิ (Hypothermia Helmet) นี้ได้ดำเนินการไปด้วยดี เนื่องจากได้รับความกรุณาจาก **ผศ.ดร.กิตติพล ชิตสกุล** อาจารย์ที่ปรึกษาโครงการ ผู้ให้คำแนะนำเป็นที่ปรึกษา ว่ากล่าว ตักเตือน ควบคุมการทำงาน ติดตามการทำงาน ให้ความรู้, เอกสารที่เกี่ยวข้อง และชี้แนะแนวทางที่เป็นประโยชน์ในการจัดทำโครงการในครั้งนี้ ทางผู้จัดทำขอขอบพระคุณ **ผศ.ดร.กิตติพล ชิตสกุล** และ **นายณรงก์ชัย สงวนพานิช** ในความสำเร็จครั้งนี้



สารบัญ

	หน้า
บทคัดย่อ	i
ABSTRACT	ii
กิตติกรรมประกาศ	iii
สารบัญ	iv
สารบัญตาราง	vii
สารบัญภาพ	ix
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 จุดมุ่งหมายในการทำโครงการ	1
1.3 ส่วนประกอบของรายงาน	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 ไข้/อุณหภูมิของคนเราสูงขึ้นได้อย่างไร	3
2.2 กลไกของการเกิดไข้	3
2.3 ลักษณะของไข้	5
2.4 ระดับของไข้	5
2.5 ระยะของการเกิดไข้	6
2.6 วิธีการวัดไข้	7
2.7 วิธีลดไข้	7
2.8 การผลิตและการระบายความร้อนในร่างกาย	9
2.9 ทฤษฎี Micro Controller	10
2.10 Sensor DS1820	15

สารบัญ (ต่อ)

	หน้า
บทที่ 3 การออกแบบ	23
3.1 การสร้างหมวกกดใช้	23
3.2 การใช้งานเซนเซอร์	24
บทที่ 4 การเขียนโปรแกรม	25
4.1 การเขียน โปรแกรมเพื่อวัดอุณหภูมิ	25
บทที่ 5 การทดลองและผลการทดลอง	29
5.1 การทดลองวัดอัตราการไหลของน้ำที่ระดับความสูงต่างๆ	29
5.2 การทดลองการวัดอุณหภูมิจากตำแหน่งต่างๆของหมวกโดยปรับระดับความสูงของน้ำ	30
5.3 การทดลองวัดกำลังของปั้มน้ำ	31
5.4 การวัดอุณหภูมิด้วยเซนเซอร์	31
5.5 การทดลองการวัดอุณหภูมิของเซนเซอร์ DS 1820	33
5.6 การทดลองการวัดอุณหภูมิของเซนเซอร์ DS 1820 กับเทอร์โมมิเตอร์จริง (Meter)	33
5.7 การทดลองการวัดค่ากระแสสูงสุดของปั้มน้ำ	34
5.8 การทดลองการทำงานของปั้มน้ำ	34
5.9 การทดลองการใช้งานของหมวก	34
5.10 การทดลองค่าอัตราการไหลของน้ำที่ค่า PWM ค่าต่างๆ	34
5.11 ผลการทดลอง	35
บทที่ 6 สรุปและวิจารณ์	50
6.1 สรุปผลการทดลองที่ได้	50
6.2 วิจารณ์ผลการทดลอง	53

สารบัญ (ต่อ)

	หน้า
6.3 ข้อผิดพลาดที่เกิดขึ้นระหว่างการทดลอง	54
หนังสืออ้างอิง	xii
ภาคผนวก	xiii



สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงการหน้าที่การทำงานของพอร์ตCแต่ละพอร์ต	15
ตารางที่ 2.2 ความสัมพันธ์ของค่าอุณหภูมิกับข้อมูลดิจิทัลเอาต์พุต	20
ตารางที่ 2.3 แสดงการแบ่งส่วนในหน่วยความจำรวมขนาด 64 บิต	20
ตารางที่ 5.1 แสดงผลการทดลองของอัตราการไหลของน้ำที่ระดับความสูงต่างๆ	35
ตารางที่ 5.2 แสดงผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ซิลิโคน	36
ตารางที่ 5.3 แสดงผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ท่อ PVC	37
ตารางที่ 5.4 แสดงผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ช่องว่างซึ่งมีฟอยล์หุ้ม	38
ตารางที่ 5.5 แสดงผลการทดลองการวัดกำลังของปั้มน้ำ	39
ตารางที่ 5.6 แสดงผลการทดลองการวัดอุณหภูมิด้วยเซนเซอร์	39
ตารางที่ 5.7 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 1 (บริเวณกลางศิรณะ)	40
ตารางที่ 5.8 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 2 (บริเวณท้ายทอย)	41
ตารางที่ 5.9 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 3 (บริเวณขมับขวา)	42
ตารางที่ 5.10 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 4 (บริเวณขมับซ้าย)	43
ตารางที่ 5.11 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น	44
ตารางที่ 5.12 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้นโดยจ่ายไฟให้ปั้มน้ำด้วย DC =12 V (อุณหภูมิน้ำ 25 °C , อัตราการไหล 250 ml/min)	45

สารบัญตาราง (ต่อ)

	หน้า
ตารางที่ 5.13 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลาโดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้นโดยจ่ายไฟให้ปั้มน้ำด้วย DC =9 V (อุณหภูมิน้ำ 25 °C , อัตราการไหล 230 ml/min)	46
ตารางที่ 5.14 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลาโดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้นโดยจ่ายไฟให้ปั้มน้ำด้วย DC =12 V (อุณหภูมิน้ำ 5 °C , อัตราการไหล 250 ml/min)	47
ตารางที่ 5.15 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลาโดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้นโดยจ่ายไฟให้ปั้มน้ำด้วย DC =9 V (อุณหภูมิน้ำ 5 °C , อัตราการไหล 230 ml/min)	48
ตารางที่ 5.16 แสดงผลการทดลองของการป้อนค่า PWM VoH = 5 V , VoL = 0 V ที่ Duty Cycle 99.5% , 75% , 50% , 25% โดยใช้ความถี่ 10 KHz 5 KHz 50Hz	49
ตารางที่ 5.17 แสดงผลการทดลองค่าอัตราการไหล ที่ Duty Cycle 99.5% , 75% , 50% , 25% โดยใช้ความถี่ 1.2 KHz (จาก PIC 16F877A)	49

สารบัญภาพ

	หน้า
รูปที่ 2.1 บล็อกไดอะแกรมการทำงานของพอร์ตA	12
รูปที่ 2.2 บล็อกไดอะแกรมการทำงานของพอร์ตB	13
รูปที่ 2.3 บล็อกไดอะแกรมการทำงานของพอร์ตC	14
รูปที่ 2.4 แสดงลักษณะตัวถังและการจัดขาใช้งานของ DS1820	16
รูปที่ 2.5 แสดงคุณสมบัติทางไฟฟ้าของ DS1820	17
รูปที่ 2.6 บล็อกไดอะแกรมภายในของ DS1820	18
รูปที่ 2.7 บล็อกไดอะแกรมการวัดค่าอุณหภูมิ	19
รูปที่ 2.8 แสดงการจัดการแหล่งจ่ายไฟเลี้ยงให้กับ DS 1820 ใน 1 - Wire	21
รูปที่ 2.9 แสดงการจัดการแหล่งจ่ายไฟภายนอกและการอินเตอร์เฟสร่วมของ DS1820	22
รูปที่ 3.1 แสดงลักษณะ โครงสร้างของหมวกคลุมใช้	23
รูปที่ 3.2 การออกแบบวงจรของส่วนวัดอุณหภูมิ	24
รูปที่ 4.1 แสดง Flow Chart ของการวัดอุณหภูมิด้วยเซนเซอร์	28
รูปที่ 5.1 แสดงชุดการทดลองอัตราการไหลของน้ำโดยไม่ใช้ปั้มน้ำ	29
รูปที่ 5.2 แสดงชุดการทดลองอัตราการไหลของน้ำโดยใช้ปั้มน้ำ	29
รูปที่ 5.3 แสดงชุดการทดลองการวัดอุณหภูมิของหมวก	30
รูปที่ 5.4 แสดงลักษณะของหมวกที่ใช้ในการทดลอง	30
รูปที่ 5.5 แสดงชุดการทดลองกำลังของปั้มน้ำ	31
รูปที่ 5.6 แสดงการต่อวงจรที่ใช้ในการวัดอุณหภูมิด้วยเซนเซอร์	31
รูปที่ 5.7 (ซ้าย – ขวา) แสดงอุปกรณ์ในการทดลองการวัดอุณหภูมิของ Sensor DS 1820	33
รูปที่ 5.8 (ซ้าย – ขวา) แสดงการทดลองการวัดค่าอุณหภูมิของเทอร์มอมิเตอร์	34
รูปที่ 5.9 แสดงกราฟผลการทดลองของอัตราการไหลของน้ำที่ระดับความสูงต่างๆ	36
รูปที่ 5.10 แสดงกราฟผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณซิลิโคน	37

สารบัญภาพ (ต่อ)

	หน้า
รูปที่ 5.11 แสดงกราฟผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณท่อ PVC	38
รูปที่ 5.12 กราฟผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณช่องว่างซึ่งมีฟอยด์หุ้ม	39
รูปที่ 5.13 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 1 (บริเวณกลางศีรษะ)	40
รูปที่ 5.14 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 2 (บริเวณท้ายทอย)	41
รูปที่ 5.15 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 3 (บริเวณขมับขวา)	42
รูปที่ 5.16 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 4 (บริเวณขมับซ้าย)	43
รูปที่ 5.17 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น	44
รูปที่ 5.18 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =12 V (อุณหภูมิน้ำ 25 °C , อัตราการไหล 250 ml/min)	45
รูปที่ 5.19 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =9 V (อุณหภูมิน้ำ 25 °C , อัตราการไหล 230 ml/min)	46
รูปที่ 5.20 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =12 V (อุณหภูมิน้ำ 5 °C , อัตราการไหล 250 ml/min)	47

สารบัญภาพ (ต่อ)

รูปที่ 5.21 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา

หน้า

48

โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้นโดยจ่ายไฟ
ให้ป้อนน้ำด้วย DC =9 V (อุณหภูมิน้ำ 5 °C , อัตราการไหล 230 ml/min)



บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ในปัจจุบัน คนส่วนใหญ่หันมาให้ความสำคัญกับสุขภาพกันมากขึ้น ทำให้เทคโนโลยีทางการแพทย์มีความเจริญก้าวหน้าไปมาก จึงทำให้เกิดการประดิษฐ์เครื่องมือเครื่องใช้ หรืออุปกรณ์ที่อำนวยความสะดวก ในการตรวจรักษา หรือวินิจฉัยโรคของผู้ป่วย เป็นจำนวนมาก อาทิเช่น เครื่องตรวจวัดคลื่นหัวใจ เครื่องวัดความดันโลหิต เครื่องสแกนสมอง เป็นต้น นอกจากนี้เครื่องมือตรวจทางการแพทย์แล้ว ก็ยังมีเครื่องอำนวยความสะดวกอื่นๆ เช่น อุปกรณ์ที่ใช้เพื่อลดไข้สำหรับผู้ป่วยที่มีไข้สูง ดังที่จะพบเห็นวางขายอยู่ในท้องตลาด ที่เรียกว่า เจลดัดไข้ แต่เนื่องจากเจลดัดไข้ นอกจากจะมีราคาที่สูงแล้ว ยังสามารถใช้เพื่อลดอุณหภูมิของคนไข้ได้เพียงแค่ชั่วคราวเท่านั้น (จะต้องนำกลับไปแช่เย็นใหม่) ทางผู้จัดทำจึงได้ทำการสร้างหมวกลดไข้ขึ้นมาเพื่อที่จะนำไปเป็นต้นแบบในการสร้างอุปกรณ์ เพื่อลดไข้ของผู้ป่วย ซึ่งจะทำให้มีความประหยัดและมีความสะดวกสบายในการใช้มากยิ่งขึ้น

1.2 จุดมุ่งหมายในการทำโครงการ

จุดมุ่งหมายในการทำโครงการนี้ เพื่อออกแบบและสร้างเครื่องมืออิเล็กทรอนิกส์สองส่วนคือ ส่วนแรกจะเป็นส่วนของหมวกลดไข้ และ ส่วนที่สองจะเป็นส่วนของ ตัววัดอุณหภูมิของหมวกลดไข้ ซึ่งหมวกลดไข่นี้จะทำขึ้นด้วยการเชื่อมต่อของสายยางซิลิโคนขนาดเล็กและข้อต่อ PVC โดยสายยางซิลิโคนจะเป็นทางเดินของของเหลวที่มีอุณหภูมิต่ำ โดยจากหลักการที่ว่าเมื่อของเหลวที่มีอุณหภูมิต่ำสัมผัสกับวัตถุที่มีอุณหภูมิสูงกว่าแล้ว จะมีการแลกเปลี่ยนอุณหภูมิเกิดขึ้น ทำให้คนไข้มีอุณหภูมิร่างกายของคนไข้ลดลงได้ และส่วนของตัววัดอุณหภูมิของหมวกลดไข้จะเป็นตัววัดค่าของอุณหภูมิของของเหลวที่ไหลผ่านหมวกลดไข่นั้น โดยจะทำหน้าที่ประมวลผลว่าอุณหภูมิของหมวกนั้นมีการกระจายตัวอย่างสม่ำเสมอหรือไม่ ซึ่งทางผู้จัดทำคิดว่าหมวกลดไข้ที่ได้จัดสร้าง

ขึ้นมาจะสามารถนำไปใช้กับผู้ป่วยได้ และในอนาคตสามารถที่จะออกแบบและพัฒนารูปแบบของหมวกให้มีความเหมาะสมอีกทั้งยังมีประสิทธิภาพในการใช้งานมากยิ่งขึ้นอีกด้วย

1.3 ส่วนประกอบของรายงาน

บทที่ 1 จะกล่าวถึงความเป็นมาและความสำคัญของการทำโครงการฉบับนี้

บทที่ 2 จะกล่าวถึง ทฤษฎีที่เกี่ยวข้อง

บทที่ 3 จะกล่าวถึงรายละเอียดในการออกแบบและสร้างหมวกลดไข้ และวงจรอิเล็กทรอนิกส์ เซนเซอร์

บทที่ 4 จะกล่าวถึงขั้นตอนการเขียนโปรแกรมเพื่อวัดอุณหภูมิโดยใช้ไมโครคอนโทรเลอร์

บทที่ 5 จะกล่าวถึงขั้นตอนการทดลอง และผลการทดลอง

บทที่ 6 จะกล่าวถึงการสรุปและวิจารณ์ผลการทดลอง ข้อผิดพลาดที่พบระหว่างการออกแบบ และขณะทดลอง



บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 ความรู้ทั่วไปเรื่องไข้

ปกติความร้อนภายในร่างกายเกิดจากการเผาผลาญสารอาหารในร่างกาย ความร้อนที่เกิดขึ้นในร่างกายจะถ่ายเทออกหลายทาง เช่น ทางลมหายใจ ระเหยออกทางเหงื่อ หรือออกไปกับปัสสาวะ ทำให้อุณหภูมิสมดุลอยู่ที่ 37 องศาเซลเซียส โดยมีศูนย์ควบคุมอุณหภูมิของร่างกาย อยู่ที่สมองส่วนที่เรียกว่า ไฮโปทาลามัส เมื่อเรามีโรคติดเชื้อหรือมีสารแปลกปลอมเข้ามาในร่างกาย สารเหล่านั้นจะไปทำให้ศูนย์ควบคุมอุณหภูมิของร่างกายทำงานเสียไป ทำให้จุดสมดุลที่ควบคุมอุณหภูมิปกติอยู่ที่ 37 องศาเซลเซียสเบี่ยงเบนไปกลายเป็น 38-39 องศาเซลเซียส ทำให้เรารู้ตัวร้อนขึ้นเรียกว่า เป็นไข้ และรู้สึกร้อนๆหนาวๆ

การหนาวสั่นเป็นการคล้ายๆ การป้องกันของร่างกายชนิดหนึ่ง เพื่อไม่ให้สูญเสียความร้อน เพราะจุดสมดุลของอุณหภูมิมันเคลื่อนที่ไปแล้ว ความรู้สึก (ที่ผิด) ของร่างกาย พยายามจะเก็บความร้อนเอาไว้โดยวิธีการหดตัวของผิวหนังทำให้เกิด อาการสั่น เพื่อให้ความร้อนมันสูงได้ระดับนั้นแต่ในขณะที่เดียวกันร่างกายก็มีอาการป้องกัน ไม่ให้ร้อนเกินไป มีการระเหยทางเหงื่อ ทางลมหายใจ ทางปัสสาวะ ก็เกิดการทำงาน 2 อย่างพร้อมๆ กัน ทำให้คนไข้รู้สึกหนาวสั่นในขณะที่ตัวร้อน

2.2 กลไกของการเกิดไข้

การเกิดไข้ เชื่อว่า มีกลไกมาจากแบคทีเรียที่ผลิตที่ออกซิน หรือเนื้อเยื่อที่ถูกทำลายปล่อยสารไพโรเจน (Pyrogen) ออกมา สารตัวนี้จะไปกระตุ้นให้เม็ดเลือดขาวปล่อยสารก่อไข้คือ เอนโดจินัสไพโรเจน (Endogenous pyrogen) ออกมา ซึ่งจะไปกระตุ้นศูนย์ควบคุมอุณหภูมิคือ ไฮโปทาลามัสให้ทำงานมากขึ้น ทำให้อุณหภูมิของร่างกายสูงขึ้น โดยทั่วไปอุณหภูมิร่างกายจะไม่สูงเกิน 41 องศาเซลเซียส (106 องศาฟาเรนไฮต์) จะคงที่อยู่ระดับหนึ่ง โดยที่ศูนย์ควบคุมความร้อนจะปรับให้อุณหภูมิไม่ขึ้นต่อไป โดยการทำให้มีเหงื่อออกมา การเกิดไข้ทำให้เกิดการเผาผลาญภายในเซลล์มากขึ้น ซึ่งก็เป็นผลดีต่อร่างกาย คือ ช่วยเพิ่มการผลิตภูมิคุ้มกันต้านทานในร่างกายและช่วยทำให้สภาพ

ร่างกายไม่เหมาะสมต่อการดำรงชีวิตของแบคทีเรีย แต่การมีไข้สูงจะทำให้เซลล์ในระบบประสาทส่วนกลางถูกทำลายเอนไซม์ต่าง ๆ ในร่างกายไม่สามารถทำงานได้ และศูนย์กลางควบคุมการหายใจอาจจะหยุดทำงาน หากอุณหภูมิสูงถึง 41 องศาเซลเซียส จะทำให้สมองถูกทำลายอย่างถาวรและหมดสติเสียชีวิตได้ ถ้าอุณหภูมิสูงกว่า 43 องศาเซลเซียส

เกิดจากระบบป้องกันของร่างกายป่าวหน่อ พอโรคเข้าสู่ร่างกายได้ ร่างกายเลยเพิ่มอุณหภูมิตัวเองเพื่อให้เชื้ออ่อนตัวลง(คงไม่ถึงกับเชื้อโรคตาย) และสามารถทำให้ระบบภูมิคุ้มกัน กำจัดโรคได้ง่ายขึ้น ไฮโปธาลามัส ซึ่งอยู่เหนือต่อมใต้สมอง (pituitary gland) ในสมองเป็นตัวควบคุมความร้อน (thermostat) มันจะตรวจจับอุณหภูมิของเลือดแล้วนำมาเปรียบเทียบกับอุณหภูมิที่กำหนดไว้ ถ้าอุณหภูมิจริง(ในเลือด) แตกต่างจากอุณหภูมิที่กำหนดไว้ มันจะพยายามเพิ่มอุณหภูมิขึ้น อย่างไรก็ตาม ไฮโปธาลามัสจะส่งสัญญาณเพื่อให้เราเหงื่อออกถ้าเรามีอุณหภูมิสูงเกินไป หรือตัวสั่นถ้าเรามีอุณหภูมิต่ำเกินไป (การทำให้เหงื่อออกจะช่วยลดอุณหภูมิร่างกายได้เมื่อเหงื่อระเหยกลายเป็นไอ ส่วนการทำให้ตัวสั่นจะช่วยทำให้ร่างกายอบอุ่นได้ เพราะว่าพลังงานที่กล้ามเนื้อใช้สั่นจะเปลี่ยนเป็นความร้อน) ในกรณีนี้ มันจะทำให้เรารู้สึกอึดอัด ซึ่งเป็นการกระตุ้นเราให้หาแหล่งความร้อนหรือความเย็นมานั่นเอง ตัวอย่างเช่น ถ้าอุณหภูมิร่างกายของคุณต่ำเกินไป คุณจะหาที่ที่อุ่นกว่าหรือซุกตัวในผ้าห่ม แต่ถ้าอุณหภูมิร่างกายสูงเกินไป คุณจะมองหาที่ที่เย็น หรือลดระดับการออกกำลังกาย (เพราะการออกกำลังกายจะผลิตความร้อน)

เมื่อคุณเริ่มมีไข้ ร่างกายคุณจะส่งสัญญาณไปยังไฮโปธาลามัสเพื่อบอกว่ามันปรับอุณหภูมิที่กำหนดไว้ให้สูงขึ้น บ่อยครั้งที่กระบวนการนี้เป็นการตอบสนองต่อเชื้อโรคที่เข้ามาในร่างกายเรา เช่น แบคทีเรียหรือไวรัส เซลล์เม็ดเลือดขาวจะตอบสนองต่อเชื้อโรคเหล่านี้ด้วยการหลั่งโมเลกุลที่เรียกว่า สารก่อไข้ (endogenous pyrogen) เมื่อสารก่อไข้เดินทางไปตามเส้นเลือดจนถึงไฮโปธาลามัส มันจะไปปรับอุณหภูมิที่กำหนดไว้ เมื่อเหตุการณ์ดังกล่าวเกิดขึ้น ไฮโปธาลามัสจะบอกกับร่างกายว่า "อุณหภูมิเย็นเกินไปนะ" จำไว้ว่า ไฮโปธาลามัสจะปรับอุณหภูมิร่างกายให้อยู่ในสภาวะปกติ (หรืออยู่ในอุณหภูมิที่ถูกกำหนดไว้) คุณจึงรู้สึกหนาวและตัวสั่นนั่นเอง

หลังจากนั้นไม่นาน เมื่ออุณหภูมิเท่ากับอุณหภูมิที่กำหนดไว้แล้ว ไฮโปธาลามัสจะส่งสัญญาณบอกว่า "อุณหภูมิปกติดีแล้วนะ" คุณจะรู้สึกหนาวได้ไม่นานและมักจะเอาผ้าห่มออกไป 1-2 ผืน เมื่ออาการไข้หาย ระดับของสารก่อไข้จะลดลง ไฮโปธาลามัสจะปรับอุณหภูมิที่กำหนดให้อยู่ในระดับเดิม ซึ่งจะทำให้อุณหภูมิร่างกายสูงกว่า คุณจะรู้สึกว่ามีเหงื่อออกและเอาผ้าห่มออกหมด หลังจากนั้น

การอุณหภูมิร่างกายจะลดลงมาอยู่ในระดับปกติ และคุณก็จะรู้สึกเป็นปกติเหมือนเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณมักจะรู้สึกหนาวจัดเมื่ออุณหภูมิเพิ่มสูงขึ้นและร้อนจัดเมื่ออุณหภูมิเริ่มลดลง มันอาจจะฟังดูแปลก แต่นั่นคือวิธีการที่ไฮโปทาลามัสปรับให้คุณชินกับการเปลี่ยนแปลงอุณหภูมิให้อยู่ระดับที่มันกำหนดไว้นั่นเอง เมื่ออุณหภูมิร่างกายไปถึงจุดที่กำหนดไว้ คุณจะรู้สึกปกติ แม้ว่าอุณหภูมิร่างกายจะสูงก็ตาม สิ่งที่คุณควรเข้าใจไม่ใช่อุณหภูมิ แต่เป็นความแตกต่างระหว่างอุณหภูมิร่างกายและอุณหภูมิที่กำหนดไว้

2.3 ลักษณะของไข้ แบ่งลักษณะของการเกิดไข้ได้ 4 ชนิด คือ

2.3.1 ไข้คงที่ (Constant fever) ไข้ชนิดนี้เนี่ยจะครบ อุณหภูมิของร่างกายนั้นจะสูงกว่าปกติตลอดคือ มากกว่า 38 องศา หรืออาจมีการเปลี่ยนแปลงเล็กน้อยชนิดนี้ แต่อุณหภูมิก็ยังอยู่ในระดับสูงกว่าปกติ

2.3.2 ไข้เว้นระยะ (Intermittent fever) เป็นลักษณะของไข้ที่อุณหภูมิสูงกว่าปกติในช่วงเวลาหนึ่งและอุณหภูมิกลับสู่ภาวะปกติในช่วงเวลา 24 ชั่วโมง เช่น ผู้ป่วยวัณโรคปอด อุณหภูมิมักจะสูงขึ้นในตอนบ่ายครับ

2.3.3 ไข้กลับ (Relapsing fever) เป็นลักษณะไข้ที่อุณหภูมิสูงกว่าปกติเป็นช่วง ๆ อาจ จะ 1-2 วัน สลับกับอุณหภูมิกลับสู่ภาวะปกติแล้วกลับสู่ภาวะไข้อีกสลับกันไป เช่น อาการของโรคมาลาเรีย

2.3.4 ไข้สูง ๆ ต่ำ ๆ (Remittent fever) ลักษณะไข้ที่อุณหภูมิสูงกว่าปกติในบางช่วง เวลาที่ไข้สูง บางช่วงไข้จะลดลงภายใน 24 ชั่วโมง แต่อุณหภูมียังสูงกว่าระดับปกติ

ถ้าไข้ต่ำ แค่วันนี้เนื้อครันตัว ไม่ต้องทานยา ก็หายเองได้ เพียงแต่พักผ่อนนอนหลับ ดื่มน้ำมาก ๆ ห่มผ้าให้แห้งออก (แต่ถ้าถึงขั้นปวดเมื่อยตามตัว ร้อนๆหนาวๆ คงไม่ใช่ไข้ต่ำแล้วละ อาการปวดเมื่อยแสดงว่าเป็น ไข้หวัดหรือเป็น ไข้หลังการใช้งานร่างกายหนัก)

2.4 ระดับของไข้

ระดับของการมีไข้สามารถแบ่งออกเป็น หลายระดับ ดังนี้

2.4.1 ไข้ต่ำ (Low fever) มีอุณหภูมิระหว่าง 37.5 องศาเซลเซียส – 38.4 องศาเซลเซียส

2.4.2 ไข้ปานกลาง (Moderate fever) มีอุณหภูมิระหว่าง 38.5 องศาเซลเซียส – 39.4 องศาเซลเซียส

2.4.3 ไข้สูง (High fever) มีอุณหภูมิระหว่าง 39.5 องศาเซลเซียส – 40.5 องศาเซลเซียส

2.4.4 ไข้สูงมาก (Hyperpyrexia) มีอุณหภูมิตั้งแต่ 40.5 องศาเซลเซียส ขึ้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ระยะของการเกิดไข้

2.5.1 ระยะเริ่มต้น เป็นช่วงที่อุณหภูมิของร่างกายกำลังขึ้นสูงถึงระดับที่ศูนย์ควบคุม อุณหภูมิถูกระงับให้ตั้งอุณหภูมิที่สูงสุดของไข้ชนิดนั้น ใช้เวลาประมาณ 2-3 นาทีถึงชั่วโมง ซึ่งทำให้อุณหภูมิสูงขึ้นประมาณ 1.1 องศาเซลเซียส – 4 องศาเซลเซียส อุณหภูมิที่สูงกว่าปกติจะเพิ่มการผลิตความร้อนในร่างกายทำให้อุณหภูมิสูงขึ้นอีก กล้ามเนื้อทำงานมากขึ้น และหลอดเลือดส่วนปลายหดตัวจากการทำงานของระบบประสาทอัตโนมัติ ถูกสั่งจากศูนย์ควบคุมอุณหภูมิ จะทำให้ความร้อนถูกระบายออกทางผิวหนังลดลง มือเท้าจะเย็น แต่ศีรษะร้อน ความร้อนจะสะสมภายในร่างกายทำให้เกิดไข้สูง(นี้แหละที่คนตั้งกระทู้ถาม) ถ้าหากอุณหภูมิที่ผิวหนังลดลงต่ำกว่า 31 องศาเซลเซียส ประสาทรับความรู้สึกเย็นที่ผิวหนังจะส่งความรู้สึกไปยังสมอง ทำให้เกิดอาการสั่นของกล้ามเนื้อ ผู้ป่วยมีอาการหนาวสั่น จะยิ่งทำให้เกิดการสร้างความร้อนเพิ่มขึ้นอีก ฉะนั้นในระยะนี้ หากผู้ป่วยมีอาการหนาวสั่น ควรห่มผ้าให้ผู้ป่วยด้วยนะครับ อาการในช่วงนี้จะพบว่าผู้ป่วยมีอาการหายใจเร็วและกระหายน้ำ เนื่องจากร่างกายขับความร้อนได้ออกไซด์ที่เกิดจากการเผาผลาญเซลล์ในร่างกายที่เพิ่มขึ้นออกไป และมีการสูญเสียน้ำจากผลของการเพิ่มอัตราการเผาผลาญ ขณะเดียวกันร่างกายก็ต้องการออกซิเจนและกลูโคสเพิ่มขึ้นเพิ่มใช้ในการเผาผลาญเซลล์ ฉะนั้นผู้ป่วยจะมีชีพจรเร็วขึ้นกว่าปกติ ด้วย

2.5.2 ระยะดำเนินของไข้ เป็นระยะที่อุณหภูมิสูงถึงระดับที่ตั้งไว้ การเพิ่มอุณหภูมิภายในร่างกาย ทำให้เลือดไหลไปที่ผิวหนังเพิ่มขึ้นเพื่อระบายความร้อนออก ทำให้ผิวหนังอุ่นจะเห็นหน้าแดง ตัวแดง อัตราการเต้นของหัวใจและการหายใจเพิ่มขึ้น จากการเพิ่มอัตราการเผาผลาญภายในเซลล์เพิ่มขึ้น ทำให้ต้องการออกซิเจนและกลูโคสเพิ่มขึ้น และร่างกายมีการสูญเสียน้ำจากการหายใจเพิ่มขึ้นทำให้กระหายน้ำ ระบบประสาทจะมีความไวต่อการถูกระงับมากขึ้น ผู้ป่วยจึงมีอาการปวดศีรษะ กลัวแสง กระสับกระส่าย บางรายอาจจะซึม ง่วงนอน หรืออาจสับสน เพ้อคลั่ง จนเกิดประสาทหลอนเลยในระยะนี้จะมีการทำลายโปรตีนจากการเพิ่มอัตราการเผาผลาญภายในเซลล์ อาจพบอัลบูมิน (albumin) ในปัสสาวะได้นะครับ(อาจนี่คือมีบั้งนะ) ผู้ป่วยมีอาการอ่อนเพลีย ปวดตามกล้ามเนื้อข้อต่อต่าง ๆ ถ้ามีไข้อยู่นาน ๆ ผู้ป่วยจะน้ำหนักตัวลดลงด้วยเนื่องจากในภาวะมีไข้ผู้ป่วยจะมีอาการเบื่ออาหาร และอาจจะมีอาการคลื่นไส้ อาเจียนด้วยในขณะที่ร่างกายมีไข้ อุณหภูมิของร่างกายไม่ได้สูงคงที่ จะมีขึ้น ๆ ลง ๆ ในขณะที่อุณหภูมิ ลดลง จะมีการระบายความร้อนออกจากร่างกายด้วยครับ โดยหลอดเลือดบริเวณผิวหนังจะขยายตัว มีเหงื่อออกมาก ร่างกายอาจเกิดภาวะขาด

น้ำได้ ถ้ามีไข้สูงนาน ๆ เนื่องจากมีการสูญเสียน้ำทางเหงื่อและทางลมหายใจ จะพบอาการผิวหนังแห้ง ริมฝีปากแตก ลิ้นเป็นฝ้า ท้องผูก ปัสสาวะออกน้อย

2.5.3 ระยะสิ้นสุดของไข้ เมื่อร่างกายสามารถกำจัดสาเหตุของไข่ออกไปแล้ว อุณหภูมิร่างกายก็จะเข้าสู่ระดับปกติโดยการทำงานของศูนย์ควบคุมอุณหภูมิ กลไกการผลิตความร้อนจะทำงาน ลดลงขณะที่กลไกการระบายความร้อนจะเพิ่มขึ้น การระบายความร้อน จะมีกลไกเช่นเดียวกับเมื่ออุณหภูมิลดลงในระยะที่ 2 อุณหภูมิอาจจะลดลงอย่างรวดเร็วภายในเวลาไม่กี่ชั่วโมง หรืออาจจะค่อย ๆ ลดลง หรือกินเวลาเป็นหลายวันก็ได้ครับ

2.6 วิธีการวัดไข้ ที่มีมาตรฐานเป็นที่ยอมรับมี 4 วิธี คือ

ทางปาก (เฉพาะเด็กโตที่ร่วมมือโดยการอมไว้ใต้ลิ้น ในเด็กเล็กทำไม่ได้เพราะอาจเสี่ยงกับการกัดปรอทแตก)

ทางทวาร (ในเด็กเล็ก)

ทางรักแร้ (ต้องระวังปรอทแก้วแตก หากเด็กคืบมาก) และ

ทางหู

เครื่องมือวัดไข้มี 3 แบบ คือ แบบปรอท แบบดิจิตอล และแบบวัดทางหู (ส่วนการใช้แถบวัดแปะที่หน้าผากหรือแบบจุกหลอกให้เด็กดูเป็นวิธีที่ไม่แม่นยำ ไม่แนะนำให้ใช้) แบบปรอทมีราคาถูก แต่ต้องใช้เวลาในการวัด เช่น หากวัดทางปากหรือทางทวารใช้เวลา 2 นาที และถ้าเด็กมีอาการคัดจมูก ทำให้ต้องหายใจทางปาก อาจต้องนานถึง 4 นาที ทางรักแร้ใช้เวลา 5 นาที และต้องระวังการแตกหักเนื่องจากเป็นแก้ว แบบดิจิตอลใช้เวลาในการวัดเพียง 10-60 วินาที แต่มีราคาแพงกว่าและต้องระวังเรื่องความแม่นยำหากเครื่องมือเบดเตอร้อ่อน แบบวัดทางหูมีราคาแพงที่สุด ใช้ในเด็กอายุ 3 เดือนขึ้นไป เด็กเล็กกว่า 3 เดือนใช้ไม่ได้เนื่องจากหูเล็กเกินไป ใช้เวลาสั้นเพียง 1-3 วินาที ถ้ามีขี้หูอุดตันจะไม่แม่นยำ ในกรณีที่ไข้ไม่สูงหรือรอได้เพื่อความแม่นยำ ควรรอประมาณ 30 นาทีให้เด็กสงบหลังการออกแรงหรือร้องไห้มาก หลังการอาบน้ำหรือหลังการทานอาหารหรือเครื่องดื่มที่ร้อนหรือเย็นจัดในกรณีที่เป็นการวัดทางปาก การแปลผลหากวัดทางทวาร ถือว่ามีไข้เมื่ออุณหภูมิมากกว่า 38°C (100.4°F) ทางปากเมื่อมากกว่า 37.5°C (99.5°F) ทางรักแร้เมื่อมากกว่า 37.3°C (99°F)

2.7 วิธีลดไข้

เป้าหมายของการลดไข้ คือ ช่วยให้เด็กสุขสบายขึ้น อาการปวดหัว ปวดตัวลดลง งอแงน้อยลง นอนหลับพักผ่อนได้ และป้องกันไม่ให้เกิดภาวะช็อกจากไข้สูงในรายที่มีความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เสี่ยง นอกจากนี้ ผู้ที่มีโรคประจำตัวเช่น โรคหัวใจ โรคปอด โรคไต โรคช็อค การมีไข้ทำให้หัวใจทำงานหนักมากขึ้น จนอาจเกิดปัญหาการทำงานของหัวใจล้มเหลว ดังนั้นจึงควรให้ความสำคัญกับการลดไข้แก่ผู้ป่วยกลุ่มนี้เป็นพิเศษ แต่ถ้าไม่มีความเสี่ยงกับภาวะช็อคจากไข้สูง หรือไม่มีโรคประจำตัวดังกล่าว หากลูกมีไข้ต่ำ ไม่มีอาการองแ่งและนอนหลับได้ดี อาจไม่จำเป็นต้องลดไข้ เพราะการบังคับให้ลูกทานยาหรือการจับเช็ดตัว อาจทำให้ร้องไห้แง่มากขึ้น

- ยาลดไข้มี 3 ประเภทคือ

1. พาราเซตามอล (Paracetamol) ขนาดยา 10-20 มก./กก./ครั้ง ทุก 4-6 ชม. ไม่มีฤทธิ์กดกระเพาะอาหาร จึงทานได้ขณะท้องว่าง ขนาดยาที่มากเกินไปเป็นพิษกับตับ คำแนะนำที่เตือนว่าไม่ควรใช้ยาต่อเนื่องเกิน 5 วัน เป็นการเตือนให้ผู้ใช้ยาควรพบหมอเพื่อหาสาเหตุของไข้ จะได้รับการรักษาที่ถูกต้องต่อไป การทานยาลดไข้ไม่ใช่การแก้ที่ต้นเหตุ เป็นเพียงบรรเทาอาการไข้เท่านั้น แต่หากได้รับการตรวจจากหมออนตราพบสาเหตุของไข้แล้ว บางครั้งการเป็นโรคบางอย่างอาจมีไข้หลายวัน การใช้ยาลดไข้ทานเกิน 5 วัน ในขนาดยาที่ถูกต้อง ก็ไม่เป็นอันตรายแต่อย่างใด

2. ไบรูโพรเฟน (Ibuprofen) หรือยาลดไข้สูง ใช้ได้ในเด็ก 6 เดือนขึ้นไป ขนาดยา 5-10 มก./กก./ครั้ง ทุก 6 ชม. หากให้ยาพาราเซตามอลแล้วไข้ไม่ลด อาจเสริมยาชนิดนี้ได้ในช่วงมื้อยาของพาราเซตามอล ในกรณีที่ยังไม่ครบ 4 ชม.จากการให้ยาครั้งสุดท้าย แต่เนื่องจากยามีฤทธิ์กดกระเพาะอาหาร จึงห้ามทานขณะท้องว่าง และทำให้เลือดออกง่าย จึงห้ามทานในรายที่มีปัญหาเลือดออกง่าย เช่น ไข้เลือดออก

3. แอสไพริน (Aspirin) ห้ามใช้ในเด็กอายุต่ำกว่า 12 ปี เพราะอาจเกิด **Reye syndrome** หากใช้ในกรณีที่ผู้ป่วยเป็นโรคไข้หวัดใหญ่ หรือโรคสุกใส (**Reye syndrome** เป็นภาวะที่อันตรายมาก เพราะจะทำให้เสียชีวิตจากตับและสมองทำงานผิดปกติอย่างรุนแรง) และทำให้มีเลือดออกง่ายเหมือนยาไบรูโพรเฟน

- การปรับอุณหภูมิห้องให้เย็นสบาย มีอากาศถ่ายเท ไม่จำเป็นต้องเปิดแอร์ แต่หากเปิด ใช้อุณหภูมิ 26-28°C ไม่เย็นเกินไปจนหนาวสั่น ใส่เสื้อผ้าที่ระบายความร้อนออกได้ง่าย ไม่ห่มผ้าหนาๆ โดยมีความเชื่อว่าถ้ามีเหงื่อออกแล้วจะทำให้ไข้ลดเร็วขึ้น เพราะจะทำให้ไข้ขึ้นสูงจนชักได้

- ดูแลให้ดื่มน้ำมากๆ เพราะหากร่างกายขาดน้ำจะทำให้ไข้สูงมากขึ้น

- การเช็ดตัว โดยผ้าชุบน้ำธรรมดา หรืออุ่นเล็กน้อย ห้ามใช้น้ำเย็นหรือแอลกอฮอล์ เพราะจะทำให้เส้นเลือดฝอยที่ผิวหนังหดตัว ระบายความร้อนออกไม่ได้ มีอาการหนาวสั่นและไข้สูงมากขึ้น เวลาเช็ดตัวให้ถอดเสื้อผ้าออกให้หมดแล้วคลุมส่วนที่ยังไม่ได้เช็ดด้วยผ้าห่มบางๆ ให้เปิดเช็ดทีละส่วน ให้เช็ดย้อนทิศทางการไหลของเลือด ไม่ควรเช็ดแบบถูแรงๆ เพราะเด็กจะเจ็บ เน้นตำแหน่งซอกคอ ไบหน้า ท้อง ข้อพับแขน หลังเข่า และวางผ้าชุบน้ำตามตำแหน่งที่เป็นซอกแขน ขาหนีบ จะช่วยดึงความร้อนออกจากเส้นเลือดได้มาก ในกรณีที่เด็กไม่ยอมให้เช็ดตัว แต่ไข้สูงมาก ให้ยาแล้วยังไม่ลดลง ให้พาเด็กอาบน้ำได้ โดยใช้วิธีให้เด็กนั่งในอ่าง แล้วใช้ฝักบัวเปิดน้ำเบาๆ รวด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

น้ำอุ่นเล็กน้อยไปตามตัว แล้วรีบเช็ดตัวให้แห้ง ควรปิดแอร์และพัดลมขณะทำการเช็ดตัวลดไข้ เพราะทำให้มีอาการหนาวสั่นได้

- การวางผ้าเย็นหรือเจลลดความร้อนบริเวณหน้าผาก ไม่ได้ช่วยให้ไข้ลดลงมาก เนื่องจากเจลช่วยลดซับความร้อนเฉพาะที่ตำแหน่งที่วางเจลหรือผ้าเย็น ไม่สามารถดึงความร้อนทั้งหมดออกจากร่างกาย แต่มีประโยชน์ช่วยบรรเทาอาการปวดศีรษะจากไข้ได้

2.8 การผลิตและการระบายความร้อนในร่างกาย

2.8.1 การผลิตความร้อน เป็นผลจากการที่มีการเผาผลาญ (Metabolism) ในร่างกาย ดังนี้

1. ฮอร์โมนไทร็อกซิน (thyroxin) ซึ่งผลิตจากต่อมไทรอยด์ ทำให้ร่างกายเพิ่มอัตราการเผาผลาญภายในเซลล์มากขึ้น มีผลให้เพิ่มความร้อนในร่างกาย
2. ฮอร์โมนนอร์อีปิเนฟรินและอีปิเนฟริน (nor-epinephrine & epinephrine) เป็นฮอร์โมนที่ผลิตมาจากต่อมหมวกไต (adrenal gland) ทำให้เพิ่มอัตราการเผาผลาญภายในเซลล์ มีผลให้เพิ่มความร้อนในร่างกาย
3. การเผาผลาญสารอาหารภายในร่างกายพวก โปรตีน คาร์โบไฮเดรต หรือไขมัน ทำให้เกิดพลังงานที่จะเพิ่มความร้อนในร่างกายอย่างน้อย 40 แคลอรี/ชั่วโมง/พื้นที่ผิวของร่างกาย หากไม่มีการระบายความร้อนออก จะทำให้อุณหภูมิในร่างกายเพิ่มขึ้น 1.1 องศาเซลเซียส (2 องศาฟาเรนไฮน์) ภายใน 1 ชั่วโมง
4. การทำงานของกล้ามเนื้อ เช่น การเคลื่อนไหว จะเกิดการเผาผลาญภายในเซลล์ได้ความร้อน 25% และความร้อนที่ร่างกายผลิตได้ทั้งหมด ถ้าหากเป็นการเคลื่อนไหวโดยการออกกำลังกาย จะมีการเผาผลาญภายในเซลล์มากขึ้น ซึ่งสามารถผลิตความร้อนได้มากถึง 50 เท่าของภาวะปกติ
5. อุณหภูมิภายในร่างกายเอง ขณะที่อุณหภูมิในร่างกายเพิ่มขึ้นจะเกิดการเผาผลาญภายในเซลล์มากขึ้น สร้างความร้อนภายในร่างกายได้ถึง 70% หากอุณหภูมิเพิ่มขึ้น 10 องศาฟาเรนไฮน์ และเมื่อมีอาการหนาวสั่นร่างกายจะสามารถผลิตความร้อนได้มากถึง 4-5 เท่าของภาวะปกติ

2.8.2 การระบายความร้อน เป็นการปรับอุณหภูมิของร่างกายให้คงที่สู่ภาวะปกติในระดับ ซึ่งขึ้นกับพื้นที่ของผิวหนังและสภาพแวดล้อม การระบายความร้อนออกจากร่างกายได้ 4 วิธี ดังนี้

1. การนำความร้อน เป็นการระบายความร้อนออกโดยมีสื่อ เช่น ของเหลว ของแข็ง แก๊ส การระบายความร้อนจากร่างกาย จะต้องสัมผัสโดยตรงกับสิ่งที่เย็นกว่า เช่น การนั่งเก้าอี้ ความร้อนจะระบายมาที่เก้าอี้

2. การพาความร้อน เป็นการระบายความร้อนโดยความร้อนจะเคลื่อนที่จากที่แห่งหนึ่ง ซึ่งร้อนกว่า ไปสู่อีกแห่งหนึ่งที่เย็นกว่าโดยอากาศรอบ ๆ ตัว เช่น การใช้พัดลม
3. การแผ่รังสี คือ การระบายความร้อนออกในรูป คลื่นแม่เหล็กไฟฟ้าจากร่างกายไปยัง สิ่งแวดล้อมที่เย็นกว่า
4. การระเหยเป็นไปไอ เป็นการระบายความร้อนออกจากร่างกายทางผิวหนังและลมหายใจ ปริมาณน้ำในร่างกายประมาณ 600-900 มิลลิลิตร จะถูกระเหยออกไป

ร่างกายสามารถระบายความร้อนออกจากร่างกายได้หลายวิธี ส่วนใหญ่จะเป็นการแผ่รังสี และ การระเหยเป็นไอ ความร้อนถูกระบายออกได้ 3 ทางคือ

1. ทางผิวหนัง จะระบายความร้อนออกได้ประมาณ 87.5% เนื่องจากร่างกายมีเนื้อที่พื้นผิวมาก และมีเลือดมาเลี้ยงมาก
2. ทางลมหายใจ จะสามารถระบายความร้อนได้ประมาณ 10.7%
3. ทางอุจจาระและปัสสาวะจะระบายความร้อนออกได้ประมาณ 1.7%

2.9 ทฤษฎี Micro Controller

PIC คือ microcontroller อีกตระกูลหนึ่ง ข้อมาจากคำว่า Peripheral Interface Controller ซึ่ง concept ของเจ้า microcontroller ตระกูลนี้ก็คือ พยายามรวมเอาทุกอย่างเอาไว้ในตัวของมัน ไม่ว่าจะเป็น PROGRAM MEMORY, RAM, EEPROM, SERIAL, I2C, PWM, A/D ฯลฯ โดยไม่จำเป็นต้องต่ออุปกรณ์เสริมจากภายนอก ในตัวของ PIC จะมีฟังก์ชันที่ใช้ในการประมวลผล รวมทั้งหน่วยความจำ ซึ่งทำให้มันเหมือนกับ CPU ตัวนี้เลยทีเดียวนะ

2.9.1 ความเร็วของ PIC

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

2.9.2 หน่วยความจำของ PIC

ในอดีตหน่วยความจำของ PIC จะค่อนข้างน้อย คืออยู่ระหว่าง 512 words ถึง 4K words แต่ในปัจจุบัน บริษัท microchip ซึ่งเป็นเจ้าของ PIC ได้พัฒนาจนทำให้ memory ของ PIC มีขนาดเป็นหลายสิบกิโลไบต์ และมีที่ท้าวางจะขยายได้ใหญ่ขึ้นเรื่อยๆ ในเรื่องของการนับขนาดของ

หน่วยความจำของ 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

2.9.3 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

สรุป concept สถาปัตยกรรมของ PIC

จะยึดถือการออกแบบที่รวบรวมทุกอย่างไว้ใน chip ตัวเดียวโดยไม่ต้องต่ออุปกรณ์ใดๆ เพิ่มเติม ผลที่ตามมาก็คือแผ่นวงจรจะมีขนาดเล็ก และอุปกรณ์ที่ใช้จะไม่มาก บางงานอาจจะใช้แค่ PIC เพียงตัวเดียวโดยไม่ต้องใช้ chip อื่นมาเพิ่มเติมเลย นี่คือคุณสมบัติพิเศษของ PIC ซึ่งปัจจุบันหลายบริษัทที่ผลิต microcontroller ก็เริ่มจะหันมาเลียนแบบแนวทางนี้ แต่ทุกอย่างย่อมมีข้อเสียเนื่องจาก concept ที่จะรวมทุกอย่างไว้ใน chip เดียว ทำให้ program memory และ data memory ไม่สามารถขยายโดยใช้กับ memory ภายนอกได้ PIC จึงเหมาะสำหรับงานเล็กๆ ไม่งานใหญ่ๆ ที่ต้องการคำนวณ และ memory เยอะๆ

PORTA ใน PIC16F87X

I/O ports บางตัวของ PIC เป็นแบบ multiplexed ซึ่ง อาจเป็นทั้ง I/O หรือ peripheral features (เช่น A/D, Serial, IIC) ซึ่งเมื่อ ขาเหล่านี้ใช้งานในส่วน peripheral ก็จะไม่สามารถใช้งานในลักษณะของ I/O ได้ PORTA และ TRISA register PORTA มีขนาด 6 bit ซึ่งเป็น port ที่เป็นได้ทั้ง Input และ Output โดยต้องเลือกแบบใดแบบหนึ่ง สามารถเลือกได้จาก register ที่มีชื่อว่า TRISA ซึ่งถ้า TRISA bit ถูก set เป็น '1' PORTA ที่มีหมายเลขบิตเดียวกันนั้นก็ทำงานเป็น input (ทำให้ port นั้นอยู่ในสถานะ hi-impedance) ส่วนถ้า TRISA bit ถูก set เป็น '0' PORTA ที่มีหมายเลขบิตเดียวกันนั้นก็ทำงานเป็น output (port จะอยู่ในสถานะ output latch) การอ่านค่า PORTA register คือการอ่านค่าสถานะของ ขา PORTA ในขณะนั้น ส่วนการเขียนค่าไปยัง PORTA คือการเขียนไปยัง latch ของ port ลักษณะการเขียนจะเป็นแบบ read-modify-write operations ซึ่งหมายความว่า ในการเขียนไปยัง port จะเริ่มด้วยการ อ่านค่า port นั้นมาก่อนแล้วทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยนแปลงค่า จากนั้นก็ทำการเขียนกลับไปยัง port latch อีกครั้งหนึ่ง ขา RA4 จะ multiplexed กับ Timer0 module clock input ซึ่งจะเรียกรวมๆ ว่า RA4/T0CKI โดยที่ ขา RA4/T0CKI จะเป็นลักษณะ Input แบบ Schmitt Trigger และ Output แบบ open drain. Port RA ทั้งหมด จะมี TTL input level และมี output แบบ full CMOS drivers ส่วน PORTA ขาอื่นๆ จะ multiplex กับ analog inputs และ Vref ของ A/D input ซึ่งการกำหนดการทำงานของแต่ละขา สามารถเลือกได้โดย clear หรือ set control bits ใน ADCON1 register

FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0 AND RA5 PINS

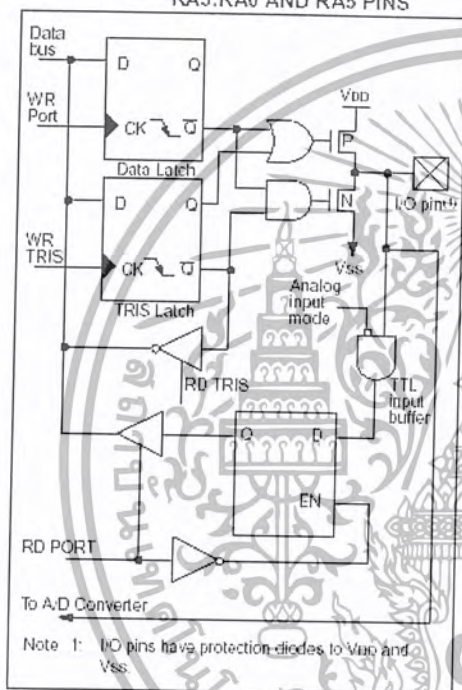
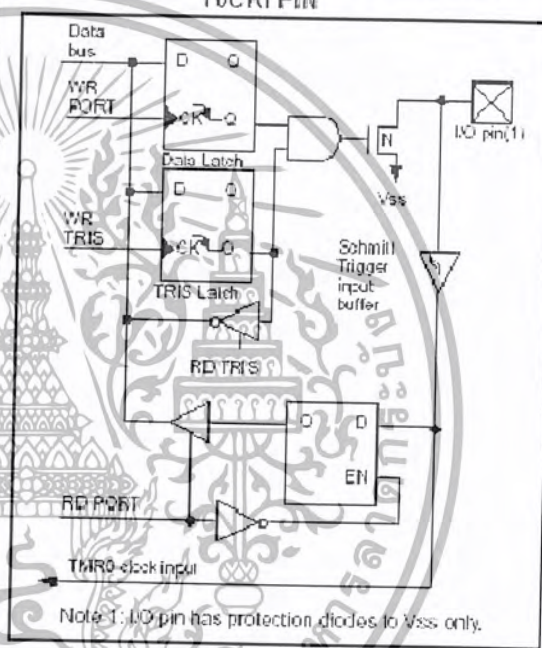


FIGURE 3-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



รูปที่ 2.1 บล็อกไดอะแกรมการทำงานของพอร์ต A

ในขณะที่เกิด Power-on Reset ขาเหล่านี้จะถูก config ให้เป็น analog input และจะอ่านค่าได้เป็น '0' TRISA register มีหน้าที่ควบคุมว่าขา PORTA ใดจะเป็น input/output ในกรณีที่ใช้ PORTA เป็น analog input TRISA register จะต้องถูก set

PORTB ใน PIC16F87X

PORTB เป็นลักษณะแบบ Port แบบสองทิศทาง ซึ่ง register ที่จะเป็นตัวกำหนดว่า port ใดจะเป็นแบบ input/output จะถูกกำหนดโดย TRISB register ถ้า set TRISB bit ใด (=1) PORTB ที่บิตนั้นก็จะเป็น input ถ้า clear TRISB bit ใด (=0) PORTB ที่บิตนั้นก็จะเป็น output ขาสามขาของ PORTB จะ multiplexed กับ Low Voltage Programming function ซึ่งได้แก่ RB3/PGM, RB6/PGC

RB7/PGD

FIGURE 3-3: BLOCK DIAGRAM OF RB3:RB0 PINS

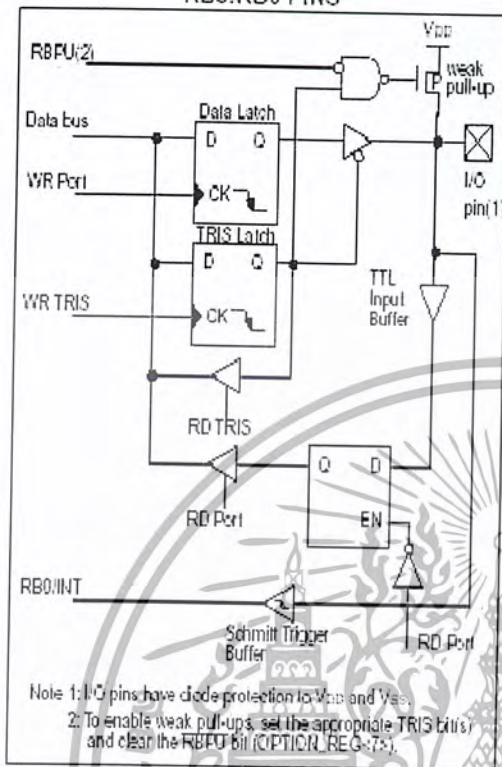
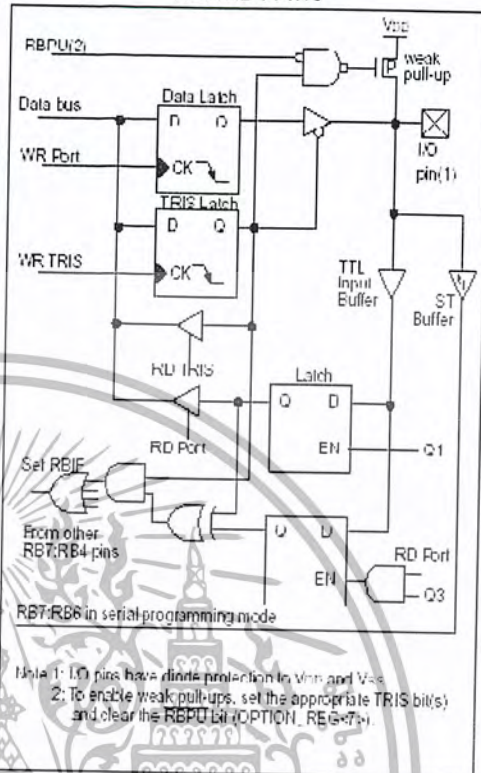


FIGURE 3-4: BLOCK DIAGRAM OF RB7:RB4 PINS



รูปที่ 2.2 บล็อกไดอะแกรมการทำงานของพอร์ตB

PORTB แต่ละ port จะมี weak pull-up อยู่ภายใน (ถ้าต้องการ pull-up แข็งๆ ต้องต่อวงจรภายนอก) เราสามารถกำหนดว่าจะใช้ pull-up ภายในหรือไม่จากการ set หรือ clear RBPU (OPTION register บิต 7) โดยถ้าเรา clear RBPU จะหมายถึงเราทำการ disable pull-up ภายใน และถ้าเรากำหนดให้ PORTB เป็น OUTPUT แล้ว pull-up จะถูก disable โดยอัตโนมัติสำหรับ PORTB นั้น ขา RB4-RB7 จะมี feature เพิ่มเติมก็คือ การกำหนดให้เกิด Interrupt เมื่อเกิดการเปลี่ยนแปลงของสถานะของสัญญาณไฟฟ้าที่ขา RB4-RB7 (โดยถ้าขาใดขาหนึ่งเกิดเปลี่ยนสถานะก็จะทำให้เกิด RB Port Change Interrupt ขึ้น ซึ่งจะทำให้ RBIF (INTCON.0) flag ถูก set โดยที่ Interrupt ประเภทนี้สามารถทำการ “wake” microcontroller จากสถานะ sleep mode ได้

RBIF flag จะถูก Clear ได้ 2 กรณี คือ

1. ทำการอ่านหรือเขียน PORTB
2. ทำการ clear RBIF flag โดยตรง

หากเราใช้ interrupt on PORTB change แล้วไม่ควรจะ enable pull-up ของ PORTB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PORTC ใน PIC16F87X

PORTC เป็นลักษณะแบบ Port แบบสองทิศทาง ซึ่ง register ที่จะเป็นตัวกำหนดว่า port ใดจะเป็นแบบ input/output จะถูกกำหนดโดย TRISC register ถ้า set TRISC bit ใด (=1) PORTB ที่บิตนั้นก็จะป็น input ถ้า clear TRISC bit ใด (=0) PORTC ที่บิตนั้นก็จะป็น output ที่ PORTC จะมีคุณสมบัติเพิ่มเติม เช่น IIC, UART, SPI, PWM, CAPTURE ขึ้นอยู่กับการเลือกใช้งาน โดยเมื่อเราทำการ enable คุณสมบัติเพิ่มเติมต่างๆ ที่ PORTC เราต้องระวังในเรื่องของการตั้งค่า TRISC ของแต่ละขาของ PORTC เพราะในการ enable คุณสมบัติบางตัวที่อยู่ที PORTC (เช่น UART) ตัวมันเองก็ จะทำการเปลี่ยน bit TRISC โดยอัตโนมัติ ดังนั้นไม่ควรที่จะตั้งค่า TRISC โดยตรงกับขาใดของ PORTC ทีทำการ enable

2.9.4 คุณสมบัติเพิ่มเติม PORTC

แต่ละ port จะ Schmitt Trigger input buffers อยู่ภายในแต่ละขา

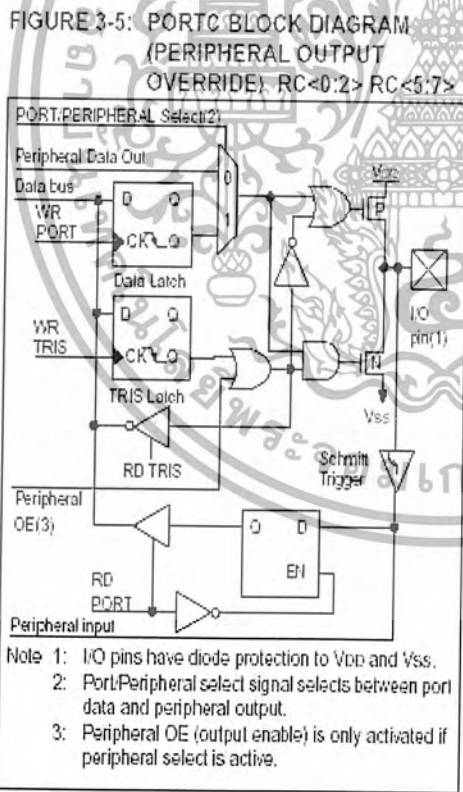
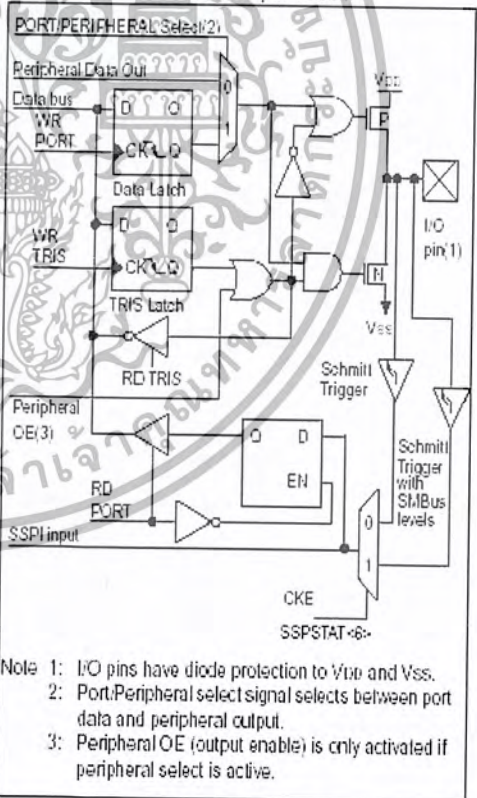


FIGURE 3-6: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<3:4>



รูปที่ 2.3 บล็อกไดอะแกรมการทำงานของพอร์ต C

ลักษณะโครงสร้างของ PORT จะแบ่งเป็น 2 กลุ่มคือ PORTC0-2,5-7 และอีกกลุ่มหนึ่งก็คือ PORTC3-4

TABLE 3-5 PORTC FUNCTIONS

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input
RC1/T1OSI/CCP2	bit1	ST	Input/output port pin or Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output
RC2/CCP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output
RC3/SCK/SCL	bit3	ST	RC3 can also be the synchronous serial clock for both SPI and I ² C modes.
RC4/SDI/SDA	bit4	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I ² C mode).
RC5/SDO	bit5	ST	Input/output port pin or Synchronous Serial Port data output
RC6/TX/CK	bit6	ST	Input/output port pin or USART Asynchronous Transmit or Synchronous Clock
RC7/RX/DT	bit7	ST	Input/output port pin or USART Asynchronous Receive or Synchronous Data

Legend: ST = Schmitt Trigger Input

TABLE 3-6 SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

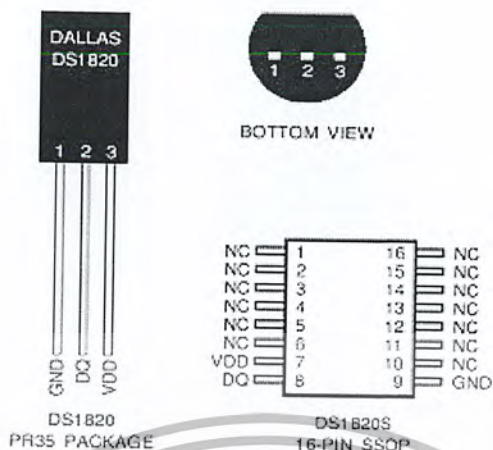
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	x.xxx xxxx	xxxxx xxxxx
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged

ตารางที่ 2.1 แสดงการหน้าที่การทำงานของพอร์ตCแต่ละพอร์ต

2.10 Sensor DS1820

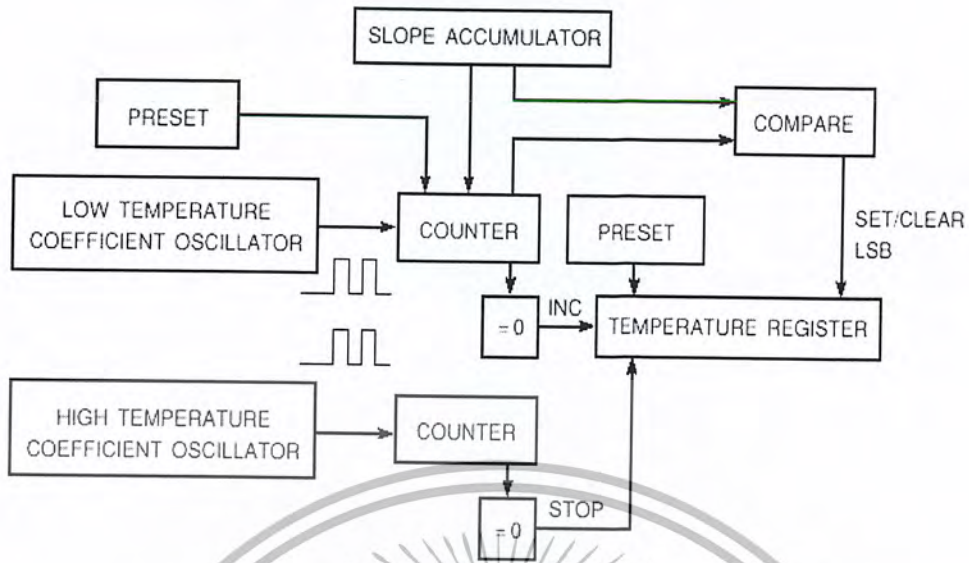
DS1820 จะให้สัญญาณเอาต์พุตออกมาเป็นแบบดิจิทัล และยังสามารถที่จะทำการโปรแกรมเข้าไปยังหน่วยความจำและควบคุมฟังก์ชันภายในไอซีได้ ซึ่งมีหน่วยความจำรวมภายในขนาด 64 บิตแบบเลเซอร์รอม ดังนั้นจึงสามารถที่จะทำการอ่านและเขียนข้อมูลต่างๆ เกี่ยวกับหน้าที่การทำงานในการตรวจวัดอุณหภูมิได้อย่างมากมายตามการประมวลผลของไมโครโปรเซสเซอร์ นอกจากนี้แล้วยังสามารถติดตั้ง DS 1820 เพื่อการตรวจวัดอุณหภูมิได้ในหลายลักษณะและหลายสถานที่ตำแหน่งการติดตั้งที่มีความแตกต่างอย่างมากกับอุปกรณ์ต่างๆ ไปไม่ว่าจะเป็นการติดตั้งภายในอาคาร, อุปกรณ์เครื่องใช้ต่างๆ หรือ ภายในเครื่องจักร และเอาต์พุตที่เป็นอนุกรมตัวเลขของ DS 1820 สามารถต่อเอาต์พุตบนสายสัญญาณเพียงเส้นเดียวได้หลายๆ ชุดโดยไม่สับสนข้อมูลซึ่งกันและกัน ในภาพที่ 2-1 แสดงรูปร่างลักษณะตัวถังและการจัดขาใช้งานของ DS1820



รูปที่ 2.4 แสดงลักษณะตัวถังและการจัดขาใช้งานของ DS1820

2.10.1 คุณสมบัติเด่นของ DS1820

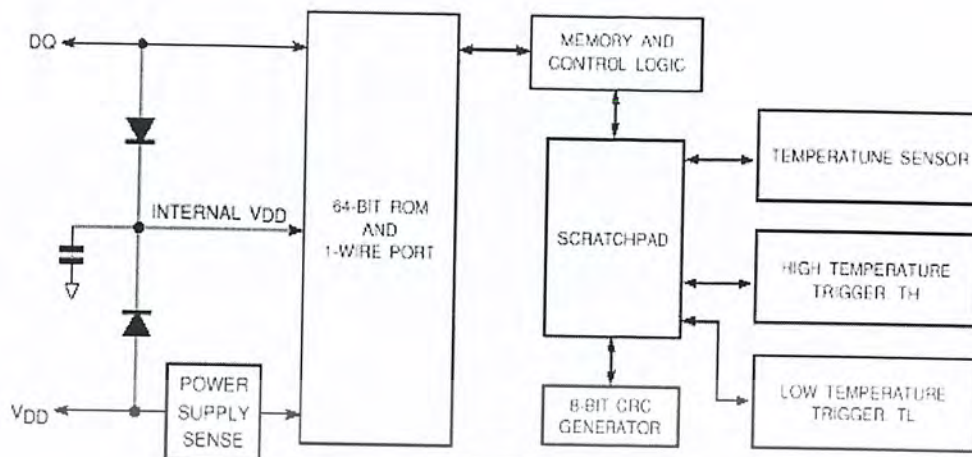
- (ก) อินเทอร์เฟซสัญญาณผ่านขาเอาต์พุตเพียงพอร์ตเดียวแบบ 1 สายข้อมูล (1 Wire™)
- (ข) ขยายจุดตรวจจับอุณหภูมิได้หลายๆ จุดบนสายข้อมูลเพียง 1 สายข้อมูล
- (ค) ไม่ต้องใช้อุปกรณ์ภายนอกมาต่อร่วม
- (ง) สามารถควบคุมการทำงานเพาเวอร์ออนได้ผ่านทางสายข้อมูล
- (จ) เพาเวอร์ขณะสแตนด์บายเป็นศูนย์
- (ฉ) ย่านการวัดอุณหภูมิตั้งแต่ -55 องศาเซลเซียสถึง +125 องศาเซลเซียสที่ 0.5 องศาต่อสแต็ป หรือตั้งแต่ย่าน -67 องศาฟาเรนไฮต์ ถึง +257 องศาฟาเรนไฮต์ที่ 0.9 องศาฟาเรนไฮต์ต่อสแต็ป
- (ช) อุณหภูมิจะถูกอ่านออกมาเป็นค่าทางดิจิตอล 9 บิต
- (ซ) อัตราความเร็วในการแปลงจากอุณหภูมิมายังเป็นค่าตัวเลขทาง ดิจิตอลเท่ากับ 200 มิลลิวินาที
- (ฌ) ผู้ใช้งานสามารถกำหนดการเซตค่าเตือนย่านอุณหภูมิได้ใน แบบ Non-Volatile
- (ญ) การเตือนย่านอุณหภูมินั้นสามารถกำหนดรหัสผ่านการสั่งการแลแอดแคสของอุปกรณ์ได้จากภายนอกพื้นที่ตรวจวัด อุณหภูมิผ่านทางโปรแกรมภายนอก
- (ฎ) เหมาะกับการประยุกต์ใช้งานตรวจวัดอุณหภูมิและติดตั้งควบคุมเทอร์โมสแตติก, ระบบโรงงานอุตสาหกรรม, ผลิตภัณฑ์, เทอร์โมมิเตอร์ หรือระบบอื่นๆ ที่มีส่วนตรวจจับอุณหภูมิทำงานร่วม



รูปที่ 2.5 แสดงคุณสมบัติทางไฟฟ้าของ DS1820

2.10.2 บล็อกไดอะแกรมภายในของ DS1820

ในภาพที่ 2-3 แสดงบล็อกไดอะแกรมส่วนประกอบของการทำงานต่างๆ ภายในตัว DS1820 มีส่วนประกอบหลักๆ 3 ส่วนด้วยกันคือ หน่วยความจำเลเซอร์รอมขนาด 64 บิต ส่วนเซ็นเซอร์อุณหภูมิและส่วนกระตุ้นเตือนอุณหภูมิแบบ non-volatile (TH และ TL) โดยอุปกรณ์ตรวจวัดอุณหภูมินี้จะถูกควบคุมสถานะการเพาเวอร์ออนและเพาเวอร์ออฟจากไลน์ข้อมูลเพียง 1 สายข้อมูลจากการเก็บรักษากำลังงานสำรองไว้ในตัวเก็บประจุภายใน ในช่วงระหว่าง คาบเวลาเมื่อสัญญาณภายในไลน์มีสถานะเป็น High และจะทำงานต่อเนื่องไปเรื่อยๆ และการหยุด การทำงานก็จะเกิดขึ้นจากการหยุดจ่ายแหล่งจ่ายในช่วงระหว่างค่าเวลานั้นเป็น Low ของไลน์ข้อมูล และจะหยุดอยู่เช่นนั้นจนกว่าขาไลน์ข้อมูลจะกลับมาเป็น High อีกครั้ง และแหล่งจ่ายไฟหลักนี้ก็จะได้จากแหล่งจ่ายไฟ +5 โวลต์ภายนอกการติดต่อข้อมูลกับ DS1820 จะติดต่อผ่านพอร์ตเพียงพอร์ตเดียวคือ Wire port ภายในพอร์ต 1-Wire นี้ในส่วน of หน่วยความจำและควบคุมฟังก์ชันโปรโตคอลของรอมจะถูกทำการเซตค่าเสียก่อน ในส่วนสำคัญของการทำงานฟังก์ชันอันดับแรกซึ่งเป็นหนึ่งในห้าลำดับของการสั่งการฟังก์ชันในรอมก็คือการอ่านหน่วยความจำรอมทำการแมตช์รอมคันทารอมกระโดดข้ามรอมเตือนการค้นหา

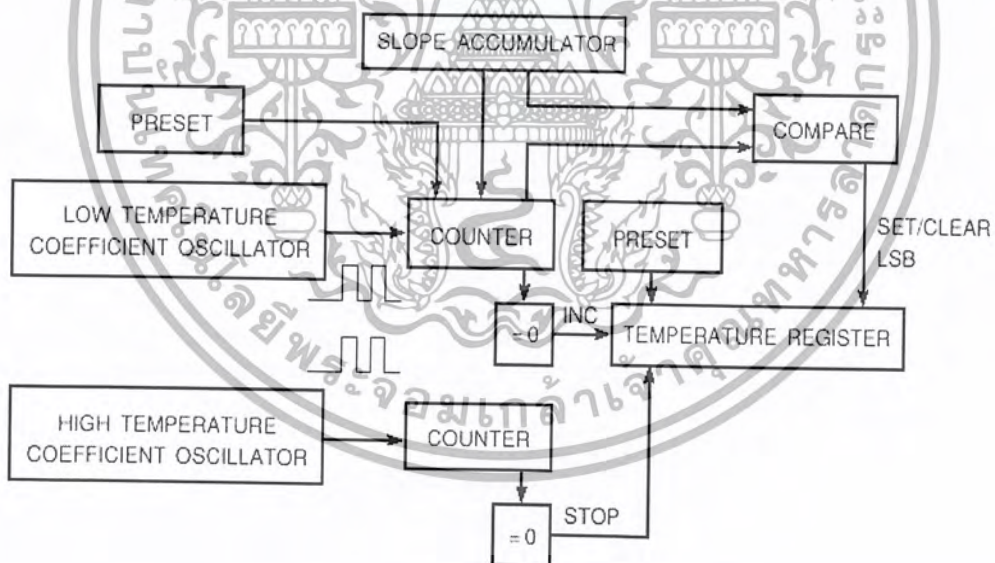


รูปที่ 2.6 บล็อกไดอะแกรมภายในของ DS1820

ซึ่งการทำงานของระบบการสั่งการนี้จะทำงานบนพื้นที่หน่วยความจำเลเซอร์รอมขนาด 64 บิต ผ่านพอร์ตไอซีแต่ละตัวและสามารถให้เอาต์พุตเดี่ยวเพื่อการกำหนดคุณสมบัติของอุปกรณ์ ตรวจสอบอุณหภูมินี้หลายๆ ตัวก็ทำได้โดยสั่งการผ่านไลน์ข้อมูล 1-wire นี้ หลังจากฟังก์ชันในรอม ถูกลำดับการทำงานแล้ว ก็พร้อมที่จะถูกใช้งาน และสามารถที่จะเข้าถึงการทำงานภายในตัวไอซีได้ทั้งหมด หน่วยความจำและส่วนควบคุมฟังก์ชันก็จะถูกเข้าถึงการทำงานได้ และส่วนจัดเก็บค่าที่เซตไว้สามารถหรืออาจจะถูกเก็บไว้ในพื้นที่ 1 ส่วนจากทั้งหมด 6 ส่วนของหน่วยความจำและส่วนควบคุมฟังก์ชันการสั่งการ ส่วนควบคุมฟังก์ชันการสั่งการหนึ่งส่วนจะถูกกำหนดคุณสมบัติของ DS1820 ให้อยู่ในรูปแบบของการวัดค่าของอุณหภูมิซึ่งผลของการวัดนี้จะถูกบันทึกไว้ใน DS1820 ในส่วนของหน่วยความจำส่วนหนึ่ง (scratchpad) และบางครั้งก็จะอ่านออกมาได้จากตารางรายชื่อของหน่วยความจำฟังก์ชันการสั่งการซึ่งเป็นการอ่านออกมาเฉพาะหัวข้อที่ถูกบันทึกไว้ในหน่วยความจำ scratchpad สัญญาณกระตุ้นเตือนค่าอุณหภูมิสูงเกินและต่ำเกิน (TH และ TL) จะประกอบด้วย 1 ไบต์ EEPROM ถ้าสัญญาณการเตือนการค้นหามิได้ถูกจ่ายเข้าไปยัง DS1820 รีจิสเตอร์เหล่านี้บางครั้งจะถูกใช้ได้อย่างทั่วๆ ไปจากหน่วยความจำที่ผู้อ่านกำหนดได้ และการเขียนเข้าไปในส่วนของการเตือน TH และ TL จะไม่ใช้หน่วยความจำฟังก์ชันการสั่งการและการเข้าไปถึงรีจิสเตอร์นี้จะอ่านผ่านหน่วยความจำ scratchpad และข้อมูลอื่นๆ ที่ต้องการอ่านและเขียนจะกระทำได้ในบิตแรกของ LSB

2.10.3 การวัดอุณหภูมิ DS1820 จะทำการวัดค่าวัดอุณหภูมิโดยอาศัยเทคนิคการวัดแบบอนบอร์ดพิเศษซึ่งเป็นเทคนิคการวัดโดยเฉพาะของอุปกรณ์ชนิดนี้ในภาพที่ 2-4 แสดงบล็อกไดอะแกรมของวัดค่าอุณหภูมิของ DS1820 ซึ่งจะอาศัยการวัดอุณหภูมิโดยการนับจำนวนรอบของสัญญาณนาฬิกาที่ออสซิลเลเตอร์ผลิตขึ้นมา ช่วงเวลาเกิดของสัญญาณนาฬิกาที่ออสซิลเลเตอร์ผลิตขึ้นมาจะ

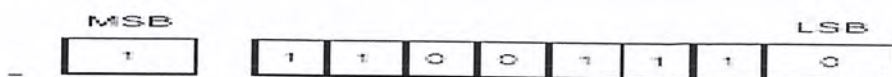
เป็นการกำหนดได้จากช่วงเวลาที่ค่าสัมประสิทธิ์อุณหภูมิต่ำสุดไปจนถึงค่าสัมประสิทธิ์อุณหภูมิสูงซึ่งจะมีค่าความถี่สัญญาณนาฬิกาที่ไม่เท่ากัน โดยที่ค่าการนับตัวเลขจะเริ่มนับที่ค่าอุณหภูมิต่ำสุดพื้นฐาน คือ -55 องศาเซลเซียส ถ้าการนับสัญญาณนาฬิกาไปถึงค่าศูนย์ก่อนที่เวลาเกิดจะเกินมา รีจิสเตอร์อุณหภูมิก็จะแสดงผลที่ค่า -55 องศาเซลเซียส ถ้าหากค่าอุณหภูมิเพิ่มขึ้น การแสดงผลของอุณหภูมิขณะนั้นก็จะสูงกว่า -55 องศาในทำนองเดียวกันนี้ การตั้งค่าของการนับจะกำหนดได้จากการเพิ่มความลาดลงของวงจรถับ ซึ่งวงจรถับนี้ต้องการการชดเชยสำหรับการแสดงคุณสมบัติของส่วนโค้งของออสซิลเลเตอร์ที่อุณหภูมิมีค่าเกินมา วงจรถับก็จะนับสัญญาณนาฬิกาอีกครั้งจนกว่ามันจะได้ค่าเป็นศูนย์ ถ้าคาบเวลาเกิดอยู่ในสภาวะสงบนิ่งไม่มีการปรับแต่งก็จะเกิดการประมวลผลใหม่อีกครั้งหนึ่งการคำนวณค่าภายใน DS1820 จะให้ความละเอียด 0.5 องศาเซลเซียสต่อสเต็ปของการเปลี่ยนแปลงอุณหภูมิการอ่านค่าอุณหภูมิจะถูกกำหนดไว้ภายใน 16 บิต โดยมีนัยสำคัญ 2 ส่วนประกอบการอ่านแสดงคุณลักษณะรายละเอียดความสัมพันธ์ของข้อมูลทางเอาต์พุตกับการจัดอุณหภูมิ ข้อมูลจะถูกส่งออกมาเป็นอนุกรมบนการอินเตอร์เฟสกับสายข้อมูล 1-wire ซึ่ง DS1820 สามารถทำการวัดค่าอุณหภูมิได้เกินย่านตั้งแต่ -55 ถึง $+125$ องศาเซลเซียสที่ 0.5 องศา



รูปที่ 2.7 บล็อกไดอะแกรมการวัดค่าอุณหภูมิ

ต่อสเต็ปค่าอุณหภูมิที่ถูกทำการปรับตั้งไว้ใน DS1820 ในเทอมของ $1/2$ องศาเซลเซียส LSB ซึ่งจะเป็นไปตามรูปแบบของข้อมูล 9 บิต

-25 องศาเซลเซียส



ที่ MSB บิตเป็นคู่เปรียบเทียบกับทุกบิตใน MSB สูงสุดของรีจิสเตอร์อุณหภูมิขนาด 2 ไบต์ ในหน่วยความจำซึ่งการอ่านค่าอุณหภูมิแบบ 16 บิต ในลักษณะสำคัญต่างๆ

2.10.4 การทำงานของสัญญาณเตือนหลังจากที่ DS1820 มีการตรวจวัดอุณหภูมิเกิดขึ้นแล้ว ค่าของอุณหภูมิก็จะทำการเปรียบเทียบเพื่อทำเป็นสัญญาณกระตุ้น การเปรียบเทียบค่าของอุณหภูมิจะเปรียบเทียบกับค่าที่ถูกบันทึกหรือกำหนดไว้ของค่าอุณหภูมิสูงสุด (TH) และค่าอุณหภูมิต่ำสุด (TL) ตลอดย่านอุณหภูมิที่วัดได้ โดยจะใช้พื้นที่รีจิสเตอร์ 8 บิต สำหรับการทำงานนี้ใน MSB ของ TH หรือ TL ที่ตรงกันก็จะถูกส่งไปยัง SB ของรีจิสเตอร์อุณหภูมิขนาด 16 บิต ถ้าผลของการวัดอุณหภูมิมิค่าสูงเกินกว่า TH หรือต่ำกว่า TL ลำดับสัญญาณเตือนภายในอุปกรณ์ก็จะถูกเซต ซึ่งลำดับของสัญญาณเตือนนี้จะถูกอัปเดตทุกครั้งที่มีการวัดค่าอุณหภูมิ เมื่อลำดับสัญญาณเตือนถูกเซต DS1820 จะมีการตอบสนองนำไปสู่การค้นหาสัญญาณเตือนการส่งการและจะยอมให้ทำการต่อ DS1820 ในลักษณะขนานกันหลายตัวได้ เพื่อทำการจำลองการวัดค่าอุณหภูมิแล้วนำมาเฉลี่ยค่าของการวัดในครั้งนั้นอีกขั้นตอนหนึ่ง



ตารางที่ 2.2 ความสัมพันธ์ของค่าอุณหภูมิกับข้อมูลดิจิทัลเอาต์พุต

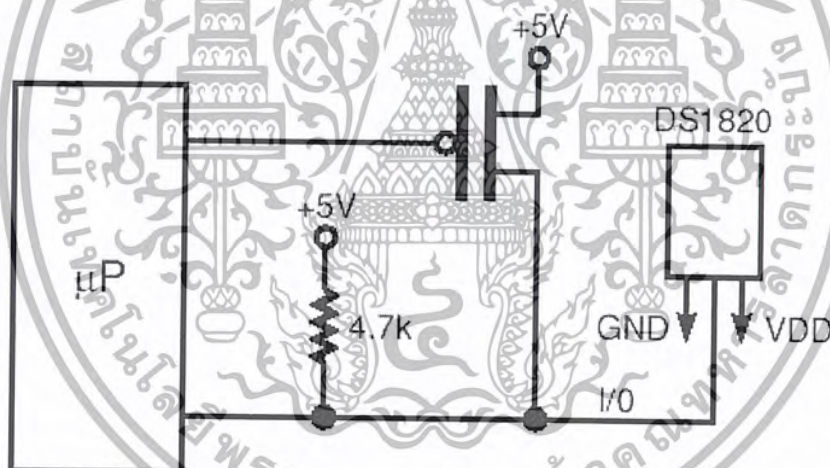
ค่าอุณหภูมิ	ดิจิทัลเอาต์พุต (Binary)	ดิจิทัลเอาต์พุต (Hex)
+125 °C	00000000 11111010	00FA
+25 °C	00000000 00110010	0032h
+1/2 °C	00000000 00000001	0001h
+0 °C	00000000 00000000	0000h
-1/2 °C	11111111 11111111	FFFFh
-25 °C	11111111 11001110	FFCEh
-55 °C	11111111 10010010	FF92h

ตารางที่ 2.3 แสดงการแบ่งส่วนในหน่วยความจำรวมขนาด 64 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.5 บิตเลขรอม

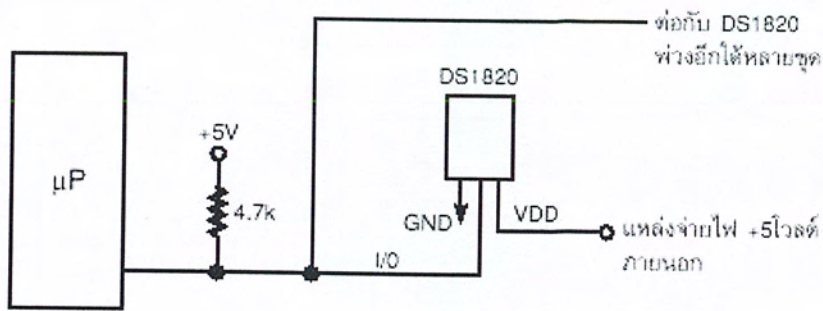
ใน DS1820 นั้นจะประกอบด้วยส่วนของรหัสหน่วยความจำรอมที่มีความยาวถึง 64 บิต โดยใน 8 บิตแรกจะเป็นรหัสตระกูล (family code) 1 – Wire ของ DS1820 (DS1820 มีรหัสเป็น 10h) และอีก 48 บิตต่อมาเป็นส่วนระบุอนุกรมตัวเลข (serial number) และอีก 8 บิตสุดท้ายคือส่วนบันทึก CRC ของ 56 บิตแรก ดังแสดงการแบ่งส่วนไว้ในหน่วยความจำรอมขนาด 64 บิต และส่วนควบคุมฟังก์ชันรอมนี้จะยอมให้ DS1820 สามารถทำงานเป็นอุปกรณ์อินเตอร์เฟสแบบ 1 – Wire ได้ และมีรายละเอียดตามโปรโตคอลของระบบบัส 1 – Wire ซึ่งฟังก์ชัน และส่วนควบคุมต่างๆใน DS1820 จะยังไม่สามารถทำงานหรือเข้าถึงได้จนกว่าจะมีการเซตค่าฟังก์ชันรอมในหน่วยความจำรอมเสียก่อน โดยในการอินเตอร์เฟสในส่วนหลักของฟังก์ชัน การสั่งการในหน่วยความจำรอมจะต้องมีลำดับฟังก์ชันดังนี้คือการอ่านหน่วยความจำรอมทำการแมตช์รอมค้นหารอม กระโดดข้ามรอม เตือนการค้นหาลงจากที่มีการเซตลำดับฟังก์ชันรอมดังกล่าวเรียบร้อยแล้ว ฟังก์ชันต่างๆ ของ DS1820 ก็จะสามารเข้าถึงได้



รูปที่ 2.8 แสดงการจัดการแหล่งจ่ายไฟเลี้ยงให้กับ DS 1820 ใน 1 – Wire

2.10.6 การต่อแหล่งจ่ายไฟ

ภาพที่ 2-6 เป็นการต่อ DS1820 ร่วมกับไมโครโปรเซสเซอร์ เพื่อการควบคุมจากระยะไกล จะสังเกตเห็นว่าที่ขารับไฟเลี้ยง VDD ของ DS1820 นั้นจะต่อกับกราวด์ แต่จะได้รับไฟเลี้ยงมาจากขา I/O ข้อมูลในแบบ 1 – Wire จากระบบควบคุมหลักไมโครโปรเซสเซอร์แทน ซึ่งวิธีนี้จะใช้สำหรับการควบคุมจากระยะไกลและไม่ต้องจัดหาแหล่งจ่ายไฟภายนอกให้กับ DS1820 ให้อย่างยากเพราะในการอินเตอร์เฟสแบบ 1 – Wire นี้สามารถที่จะทำการกำหนดการทำงาน (Power On) ของไอซีให้จ่ายไฟเลี้ยงวงจรภายในตัว ไอซีแทนเป็นการจัดแหล่งจ่ายไฟภายนอกให้



รูปที่ 2.9 แสดงการจัดแหล่งจ่ายไฟภายนอกและการอินเตอร์เฟสร่วมของ DS1820

DS1820 ณ จุดที่ติดตั้งใช้งาน และการส่งการจากไมโครโปรเซสเซอร์ ยังทำงานได้เหมือนเดิมและสามารถที่จะต่อ DS1820 ขนานกันโดยใช้การอินเตอร์เฟสผ่านบัส 1 – Wire เดียวกันได้หลายตัว จึงเหมาะกับการตรวจวัดอุณหภูมิหลายจุดที่ควบคุมได้จากระยะไกลผ่านระบบไมโครโปรเซสเซอร์



บทที่ 3

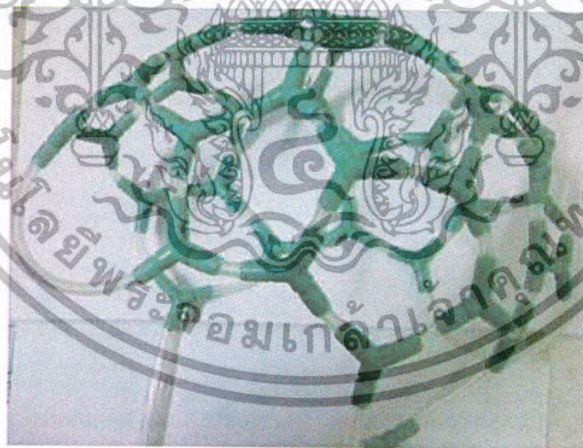
การออกแบบ

3.1 รายละเอียดหมวกลดไข้

อุปกรณ์ที่ใช้ทำหมวกลดไข้

1. สายยางซิลิโคน ขนาดเส้นผ่านศูนย์กลาง 4 มิลลิเมตร
2. ข้อต่อสายยาง PVC
3. ฟอยล์กระจายความร้อน
4. ป้อน้ำขนาด 12V DC แรงดันน้ำ 1.20 เมตร

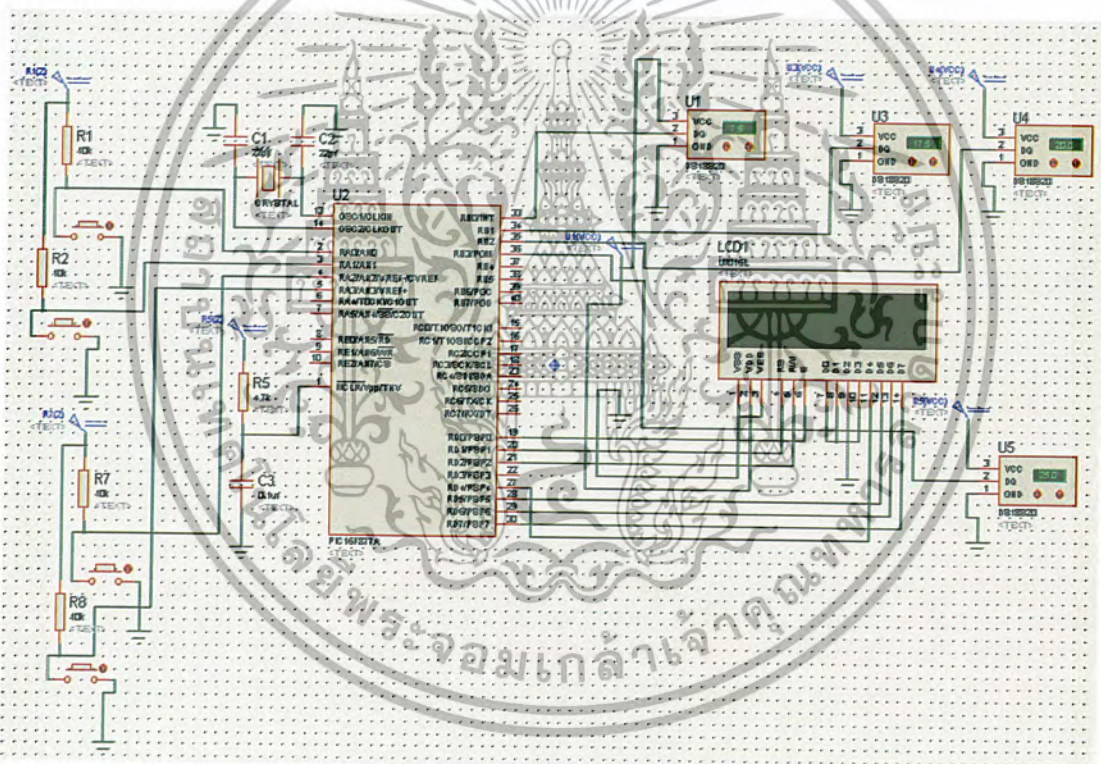
หมวกลดไข้นี้จะทำขึ้นด้วยการเชื่อมต่อของสายยางซิลิโคนขนาดเล็กและข้อต่อ PVC โดยสายยางซิลิโคนจะเป็นทางเดินของของเหลวที่มีอุณหภูมิต่ำ โดยจากหลักการที่ว่าเมื่อของเหลวที่มีอุณหภูมิต่ำสัมผัสกับวัตถุที่มีอุณหภูมิสูงกว่าแล้ว จะมีการแลกเปลี่ยนอุณหภูมิเกิดขึ้น ทำให้คนไข้มีอุณหภูมिर่างกายของคนไข้ลดลงได้ หมวกลดไข้จะมีลักษณะดังรูป



รูปที่ 3.1 แสดงลักษณะ โครงสร้างของหมวกลดไข้

3.2 การใช้งานเซนเซอร์

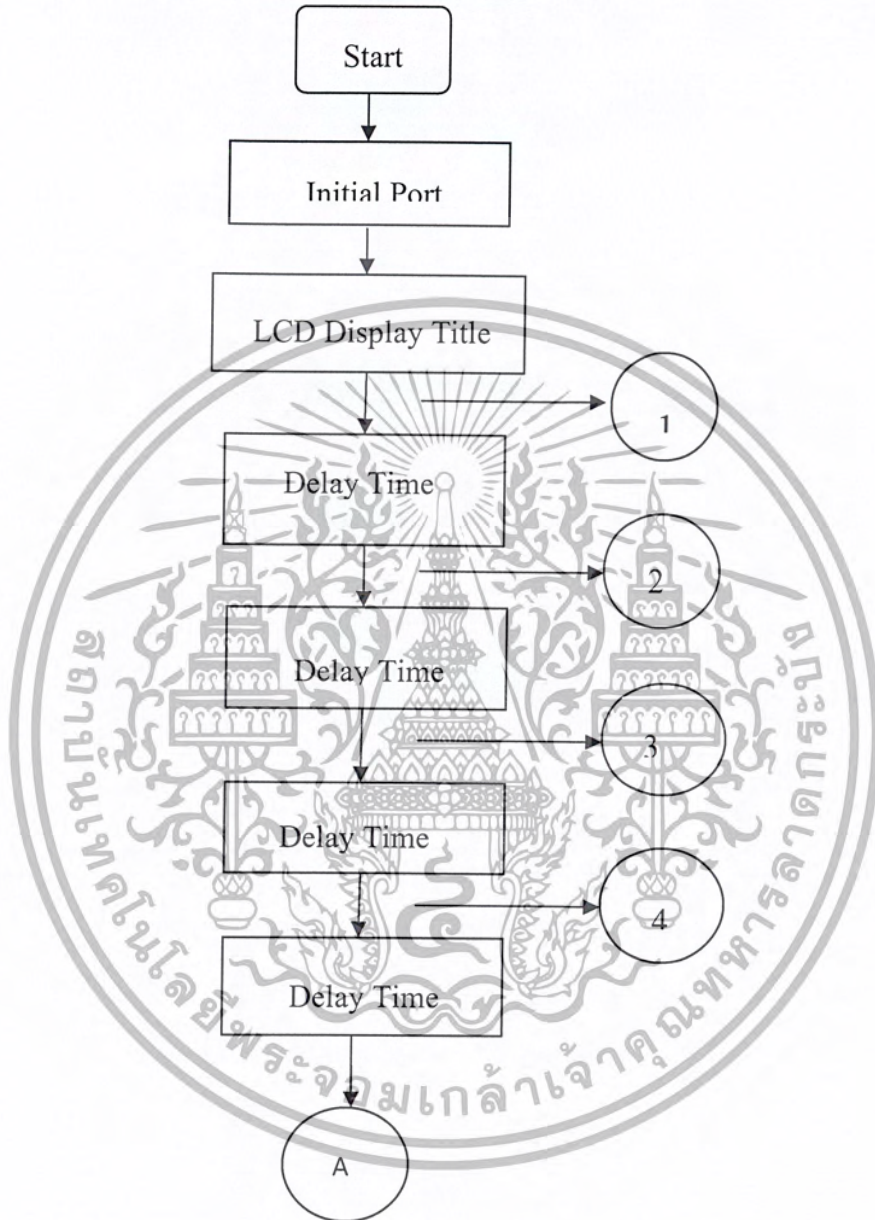
การออกแบบของโครงการนี้จะใช้ไมโครโปรเซสเซอร์เบอร์ 16F877A ในการประมวลผล ส่วนการวัดอุณหภูมิจะใช้เซนเซอร์อุณหภูมิที่เป็นแบบดิจิตอลเบอร์ DS18B20 ในการวัดอุณหภูมิและในส่วนแสดงผลจะใช้ LCD ขนาด 16 ตัวอักษร 2 บรรทัด ในการแสดงผลโดยโครงการนี้ได้ออกแบบ โดยให้เซนเซอร์วัดค่าอุณหภูมิที่ต้องการแล้วส่งค่าเพื่อให้ไมโครโปรเซสเซอร์เบอร์ 16F877A ประมวลผลแล้วแสดงค่าอุณหภูมิออกทาง LCD ขนาด 16 ตัวอักษร 2 บรรทัด โดยเซนเซอร์ที่ใช้วัดจะมี 4 ตัวด้วยกันแต่แต่ละตัวจะถูกติดตั้งในหมวกที่ตำแหน่งต่าง ๆ กัน 4 จุด นั่นคือ 1. ส่วนหน้าผาก 2. ส่วน ด้านหลังศีรษะ 3. ศรีษะด้านซ้าย 4. ศรีษะด้านขวา และจะแสดงอุณหภูมิแต่ละจุด โดยมีสวิทซ์ 4 ตัวเป็นตัวเลือกว่าจะวัดอุณหภูมิจากเซนเซอร์ตัวไหน

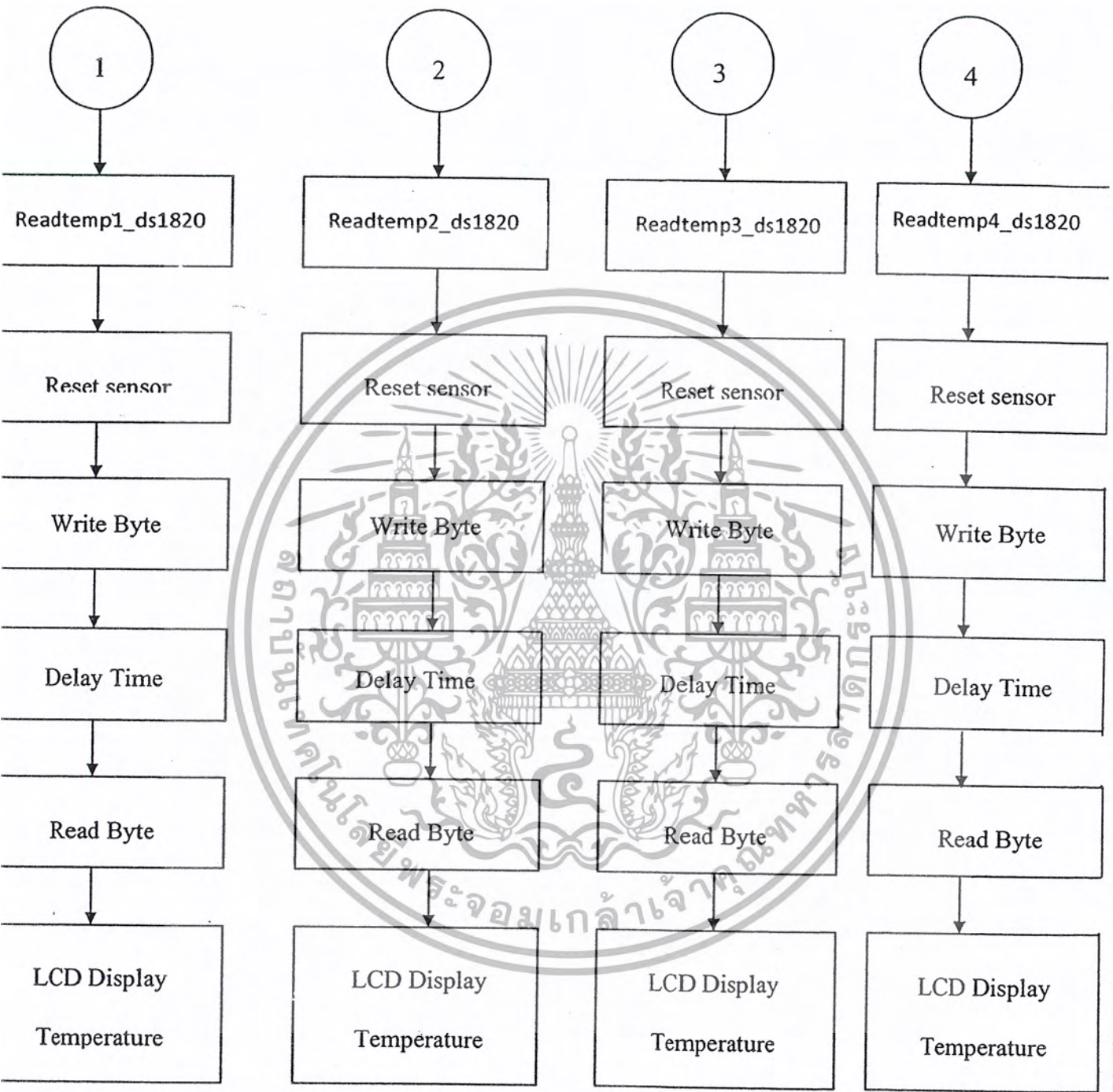


รูปที่ 3.2 การออกแบบวงจรของส่วนวัดอุณหภูมิ

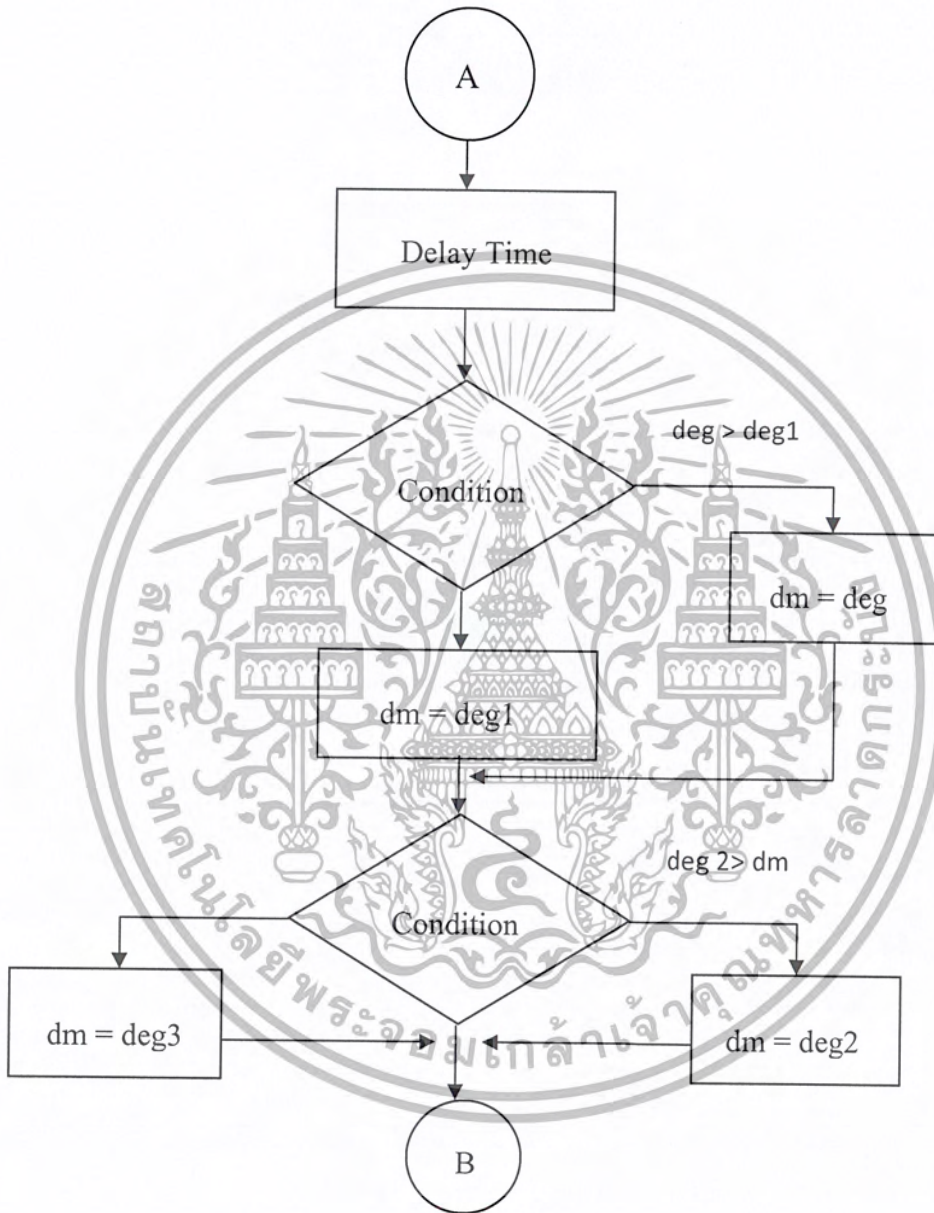
บทที่ 4

การเขียนโปรแกรม

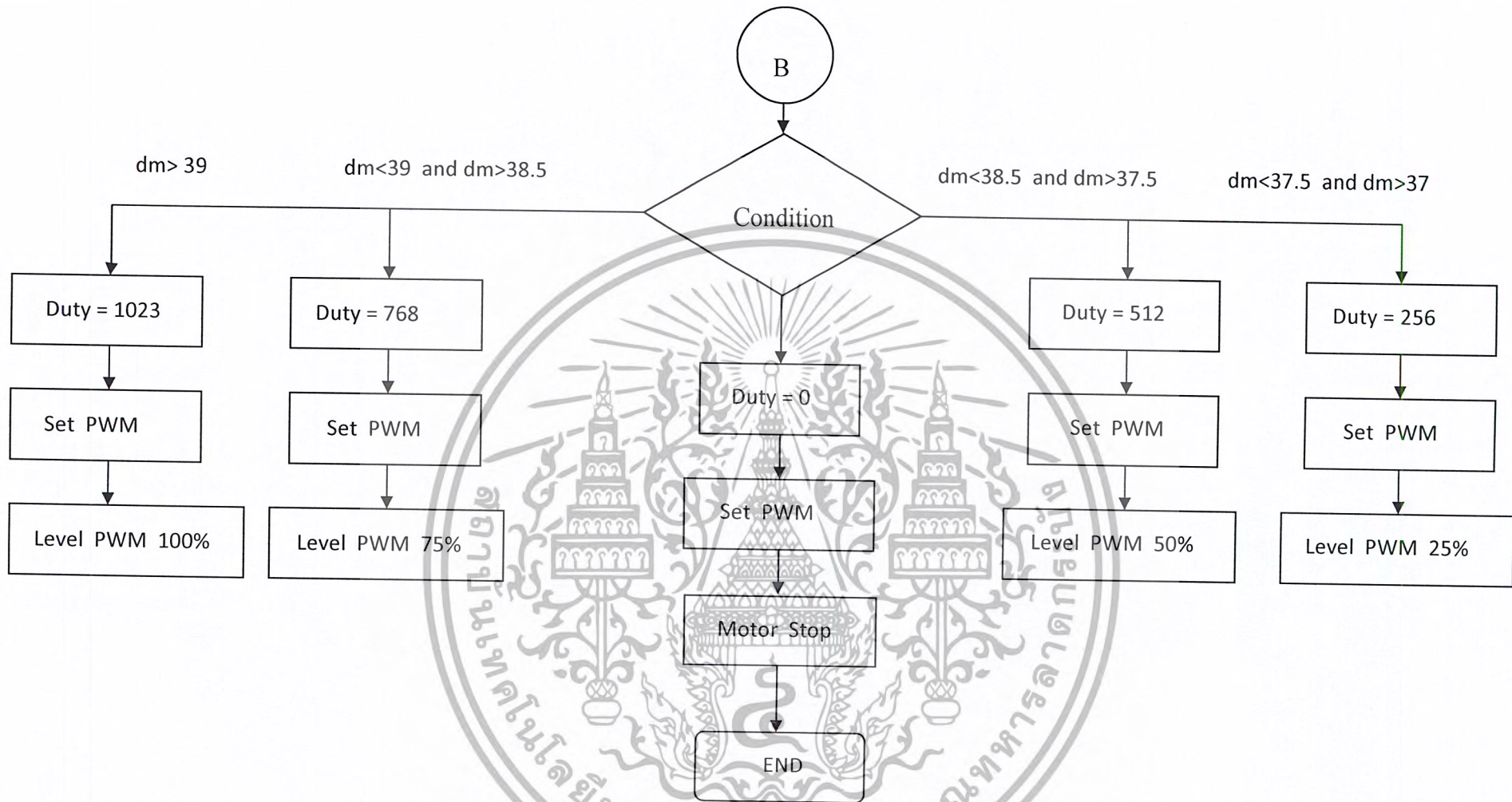




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 แสดง Flow Chart ของการวัดอุณหภูมิด้วยเซ็นเซอร์

บทที่ 5

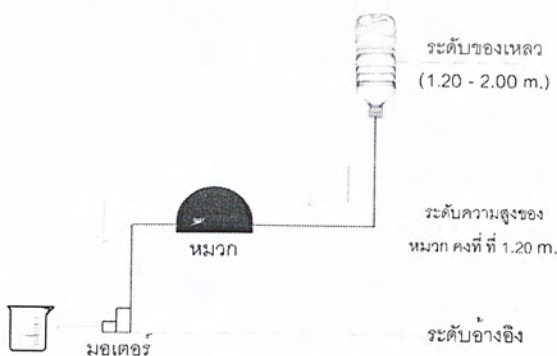
การทดลอง และผลการทดลอง

5.1 การทดลองวัดอัตราการไหลของน้ำที่ระดับความสูงต่างๆ

1. **ไม่มีมอเตอร์** ทดลองโดยการวัดอัตราการไหลของน้ำโดยการยกระดับน้ำให้สูงขึ้น จาก ระดับ 1.2 m และเพิ่มระดับความสูงที่ละ 0.1 m และยกสูงขึ้นเรื่อยๆจนระดับน้ำสูงสุดที่ 2.0 m โดยจัดชุดการทดลองดังรูปที่ 4



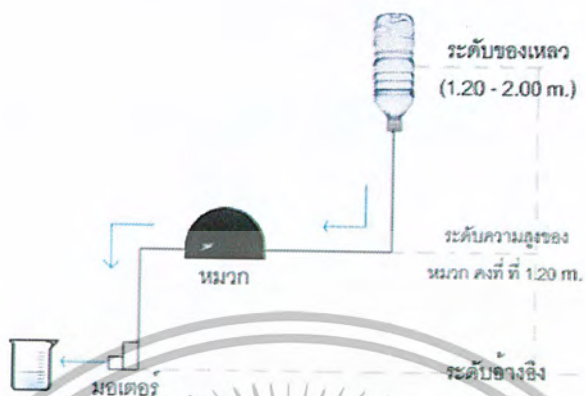
2. **มีมอเตอร์** ทดลองโดยการวัดอัตราการไหลของน้ำโดยการยกระดับน้ำให้สูงขึ้น จาก ระดับ 1.2 m และเพิ่มระดับความสูงที่ละ 0.1 m และยกสูงขึ้นเรื่อยๆจนระดับน้ำสูงสุดที่ 2.0 m แต่ใช้ปั้มน้ำเป็นตัวช่วยดึงน้ำออกจากหมวก ดังแสดงในรูปที่ 5



รูปที่ 5.2 แสดงชุดการทดลองวัดอัตราการไหลของน้ำโดยใช้ปั้มน้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การทดลองการวัดอุณหภูมิจากตำแหน่งต่างๆของหมวกโดยปรับระดับความสูงของน้ำ



รูปที่ 5.3 แสดงชุดการทดลองการวัดอุณหภูมิของหมวก



รูปที่ 5.4 แสดงลักษณะของหมวกที่ใช้ในการทดลอง

การทดลอง

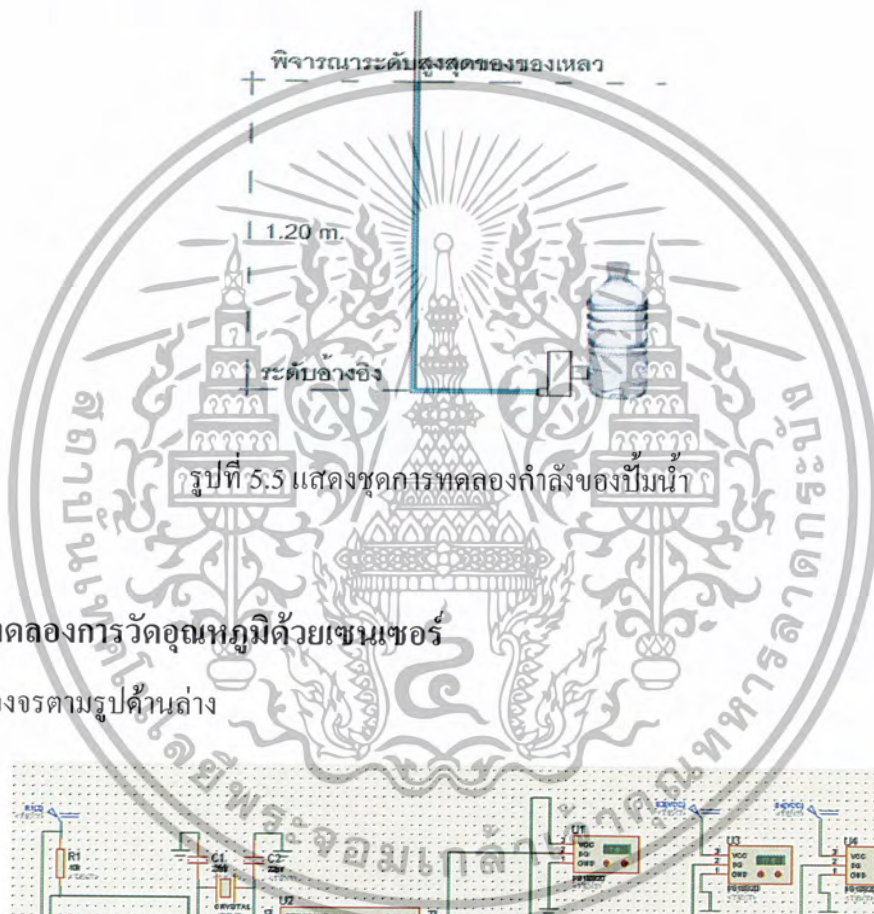
1. จัดอุปกรณ์ดังรูปที่ 6 โดยให้ระดับหมวก สูงจากพื้น 1.20 m
2. เริ่มต้น โดยการให้ขวน้ำสูงที่ระดับ 1.20 m
3. วัดอุณหภูมิของน้ำที่ไหลผ่านหมวกที่บริเวณ ซิลิโคน
4. บันทึกผลการทดลอง
5. กลับไปทำตั้งแต่ข้อ 2.- 4. โดยปรับระดับของขวน้ำให้สูงขึ้นทีละ 0.10 m จนถึงที่ตำแหน่ง ความสูง 2.0 m

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ทำการทดลองเหมือนเดิมตั้งข้อ 2.-5. โดยเปลี่ยนตำแหน่งการวัดจาก ซิลิโคน เป็น PVC และ ฟลอยด์ ตามลำดับ

5.3 การทดลองวัดกำลังของปั้มน้ำ

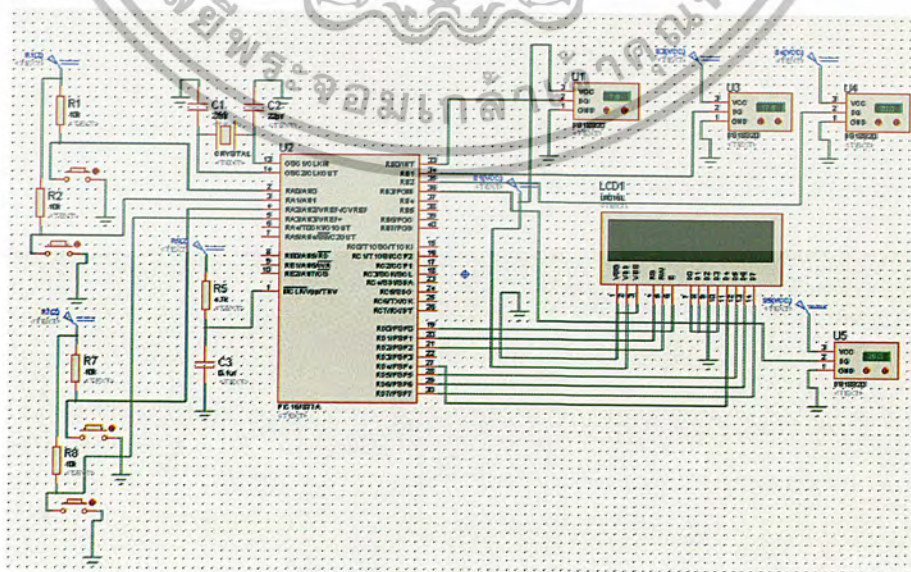
การทดลองนี้จะทดสอบโดยการให้ปั้มน้ำทำการปั้มน้ำให้ขึ้นไปบนซิลิโคนในแนวตั้งแล้ววัดระดับสูงสุดที่ปั้มน้ำสามารถปั้มน้ำขึ้นไปได้



รูปที่ 5.5 แสดงชุดการทดลองกำลังของปั้มน้ำ

5.4 การทดลองการวัดอุณหภูมิด้วยเซนเซอร์

1.ต่อวงจรตามรูปด้านล่าง



รูปที่ 5.6 แสดงการต่อวงจรที่ใช้ในการวัดอุณหภูมิด้วยเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หลังจากต่อวงจรเสร็จแล้วนำเซนเซอร์วัดอุณหภูมิตัวที่ 1 2 3 และ 4 มาวัดอุณหภูมิไอน้ำแข็ง โดยนำเซนเซอร์มาจุ่มลงในถุงน้ำแข็งโดยมีถุงพลาสติกเป็นตัวกั้นแล้วกดสวิทช์เพื่อเลือกที่จะวัดค่าจากเซนเซอร์ตัวไหนแล้วสังเกตค่าที่ได้บนจอแสดงผลLCDแล้วบันทึกค่าที่ได้ลงในตารางทำเช่นนี้จนครบทั้ง 4 ตัว

3. นำเทอร์โมมิเตอร์มาตรฐานมาวัดอุณหภูมิไอน้ำแข็งโดยจุ่มลงในถุงน้ำแข็งโดยมีถุงพลาสติกเป็นตัวกั้นแล้วบันทึกผลลงในตาราง

4. นำเซนเซอร์วัดอุณหภูมิตัวที่ 1 2 3 และ 4 มาวัดอุณหภูมิห้อง โดยกดสวิทช์เพื่อเลือกที่จะวัดค่าจากเซนเซอร์ตัวไหนแล้วสังเกตค่าที่ได้บนจอแสดงผลLCDแล้วบันทึกค่าที่ได้ลงในตารางทำเช่นนี้จนครบทั้ง 4 ตัว

5. นำเทอร์โมมิเตอร์มาตรฐานมาวัดอุณหภูมิห้องแล้วบันทึกผลลงในตาราง

6. นำเซนเซอร์วัดอุณหภูมิตัวที่ 1 2 3 และ 4 มาวัดอุณหภูมิน้ำที่อุณหภูมิห้อง โดยนำเซนเซอร์จุ่มลงในน้ำธรรมดาแล้วกดสวิทช์เพื่อเลือกที่จะวัดค่าจากเซนเซอร์ตัวไหนและสังเกตค่าที่ได้บนจอแสดงผลLCDแล้วบันทึกค่าที่ได้ลงในตารางทำเช่นนี้จนครบทั้ง 4 ตัว

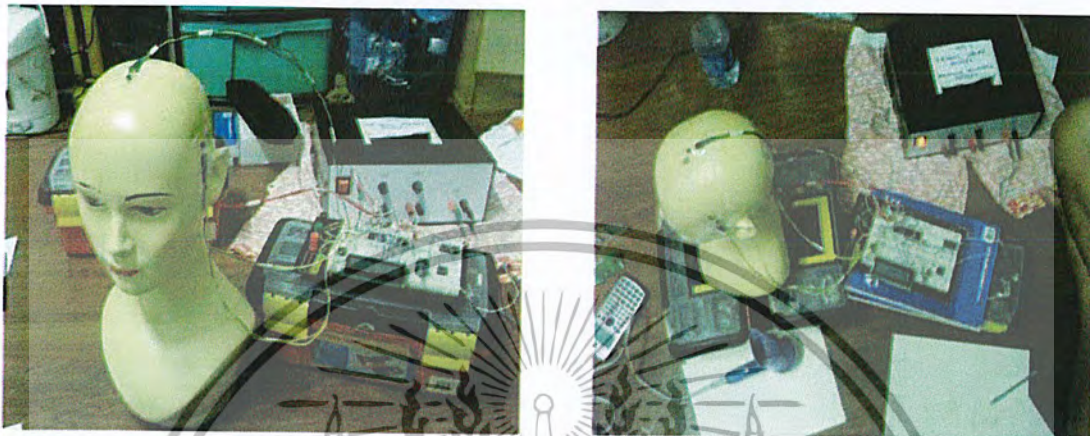
7. นำเทอร์โมมิเตอร์มาตรฐานมาวัดอุณหภูมิน้ำธรรมดาที่อุณหภูมิห้องแล้วบันทึกผลลงในตาราง

8. นำเซนเซอร์วัดอุณหภูมิตัวที่ 1 2 3 และ 4 มาวัดอุณหภูมิน้ำอุ่น โดยนำเซนเซอร์จุ่มลงในน้ำอุ่นแล้วกดสวิทช์เพื่อเลือกที่จะวัดค่าจากเซนเซอร์ตัวไหนและสังเกตค่าที่ได้บนจอแสดงผลLCDแล้วบันทึกค่าที่ได้ลงในตารางทำเช่นนี้จนครบทั้ง 4 ตัว

9. นำเทอร์โมมิเตอร์มาตรฐานมาวัดอุณหภูมิน้ำอุ่นแล้วบันทึกผลลงในตาราง

5.5 การทดลองการวัดอุณหภูมิของเซนเซอร์ DS 1820

การทดลองนี้จะให้ความร้อนเข้าไปในหัวหุ่นจนอุณหภูมิถึง 40°C แล้วจับเวลาจากการลดลงของอุณหภูมิของทั้ง 4 จุด โดยใช้ SENSOR DS 1820 วัดบริเวณ 1. กลางศีรษะ 2. ท้ายทอย 3. ขมับซ้าย 4. ขมับขวา



รูปที่ 5.7 (ซ้าย – ขวา) แสดงอุปกรณ์ในการทดลองการวัดอุณหภูมิของ Sensor DS 1820

5.6 การทดลองการวัดอุณหภูมิของเซนเซอร์ DS 1820 กับ เทอร์มอมิเตอร์จริง (Meter)

1. เป่าหุ่นจำลองที่ติดปั๊มน้ำ และ Sensor บริเวณกลางศีรษะ ให้อุณหภูมิถึง 40°C และปล่อยให้เย็นลง

ตามธรรมชาติ แล้วบันทึกเวลาในการลดลงของอุณหภูมิในแต่ละช่วง

2. ทำเช่นเดียวกับข้อ 1. แต่

- 2.1 จ่ายไฟให้ปั๊มน้ำด้วย $\text{DC} = 12\text{ V}$ (อุณหภูมิน้ำ 25°C , อัตราการไหล 250 ml/min)

- 2.2 จ่ายไฟให้ปั๊มน้ำด้วย $\text{DC} = 9\text{ V}$ (อุณหภูมิน้ำ 25°C , อัตราการไหล 230 ml/min) แล้ว

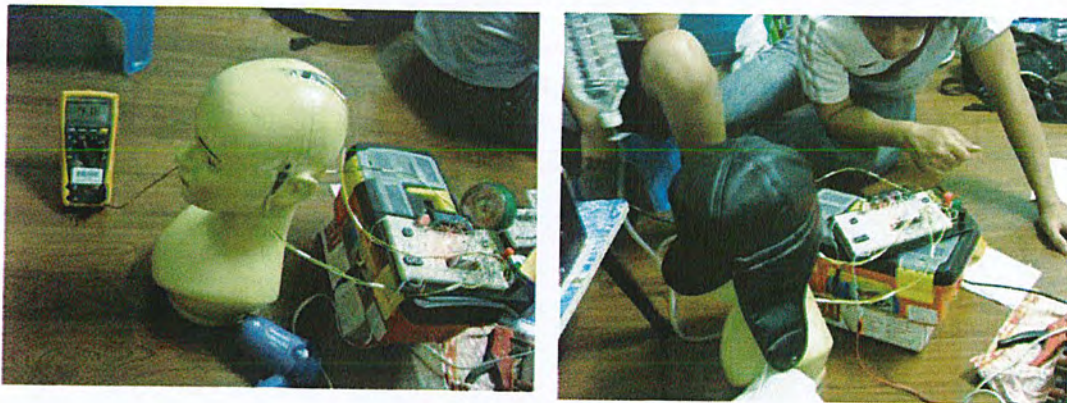
Plot กราฟ

3. ทำเช่นเดียวกับข้อ 1. แต่

- 3.1 จ่ายไฟให้ปั๊มน้ำด้วย $\text{DC} = 12\text{ V}$ (อุณหภูมิน้ำ 5°C , อัตราการไหล 250 ml/min)

- 3.2 จ่ายไฟให้ปั๊มน้ำด้วย $\text{DC} = 9\text{ V}$ (อุณหภูมิน้ำ 5°C , อัตราการไหล 230 ml/min) แล้ว Plot

กราฟ



รูปที่ 5.8 (ซ้าย – ขวา) แสดงการทดลองการวัดค่าอุณหภูมิของเทอร์มอมิเตอร์

5.7 การทดลองการวัดค่ากระแสสูงสุดของปั้มน้ำ

โดยการวัดจากเครื่องมือ พบว่า ขณะมีน้ำกินไฟ 1.65 A (100% DC)

ขณะไม่มีน้ำกินไฟ 0.5 A (100% DC)

อ้างอิงจาก Digital Clamp Amp (kewTCH KT 203)

5.8 การทดลองการทำงานของปั้มน้ำ

วัดอัตราการไหลที่กำลังสูงสุดของปั้มน้ำ 12V โดยใช้ปั้มน้ำ 1 ตัว (ปั้มน้ำเข้า) ได้อัตราการไหล 2050 ml/min

วัดอัตราการไหลที่กำลังสูงสุดของปั้มน้ำ 12V โดยใช้ปั้มน้ำ 2 ตัว (ปั้มน้ำเข้าและคังน้ำออก) ได้อัตราการไหล 340 ml/min

5.9 การทดลองการใช้งานของหมวก

ทดลองนำหมวกทดสอบอุณหภูมิมาครอบที่ศีรษะของผู้ทำการทดลอง โดยใช้น้ำอุณหภูมิประมาณ 5 °C

5.10 การทดลองค่าอัตราการไหลของน้ำที่ค่า PWM ค่าต่างๆ

1. ทดลองโดยป้อนค่า PWM $V_{OH} = 5 \text{ V}$, $V_{OL} = 0 \text{ V}$ ที่ Duty Cycle = 99.5% , 75% , 50% , 25% โดยใช้ความถี่ 10 KHz 5 KHz 50Hz

2. ทดลองโดยป้อนค่า PWM ที่ Duty Cycle 99.5% , 75% , 50% , 25% โดยใช้ความถี่ 1.2 KHz (จาก PIC 16F877A)

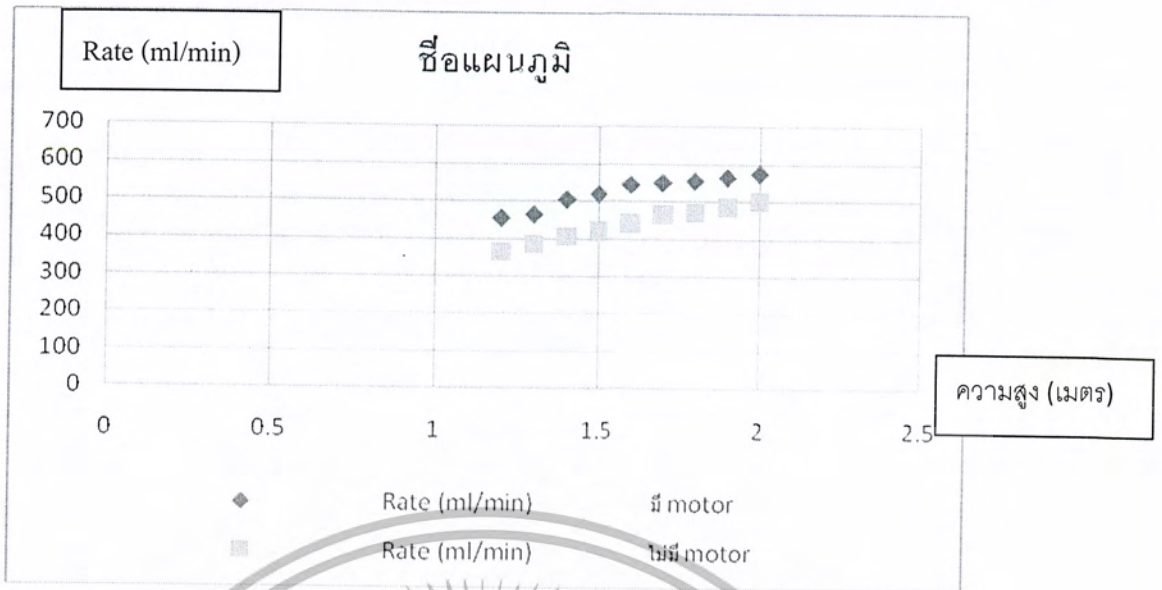
5.11 ผลการทดลอง

ผลการทดลองการวัดอัตราการไหลของน้ำที่ระดับความสูงต่างๆทั้งแบบที่มีปั๊มกับแบบไม่มีปั๊ม ได้ค่าอัตราการไหลดังตารางที่ 1 และการวัดอุณหภูมิของน้ำที่ไหลผ่านหมวกในตำแหน่งต่างๆ โดยวัดตามระดับความสูงที่มีการปรับระดับ ตั้งแต่ 1.2, 1.3, ถึง 2.0 m ได้ค่าอุณหภูมิตามตารางที่ 2 ถึง ตารางที่ 4 และผลการทดลองของการวัดกำลังของปั้มน้ำดังแสดงในตารางที่ 5 และผลการทดลองการวัดอุณหภูมิโดยใช้เซนเซอร์ 4 ตัว ได้ผลตามตารางที่ 6

ตารางที่ 5.1 แสดงผลการทดลองของอัตราการไหลของน้ำที่ระดับความสูงต่างๆ

ความสูง	Rate (ml/min)	
	มี motor	ไม่มี motor
1.2	455	365
1.3	465	385
1.4	505	405
1.5	520	420
1.6	545	440
1.7	550	465
1.8	555	470
1.9	565	485
2.0	575	500

จากค่าอัตราการไหลในตารางที่ 5.1 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.9

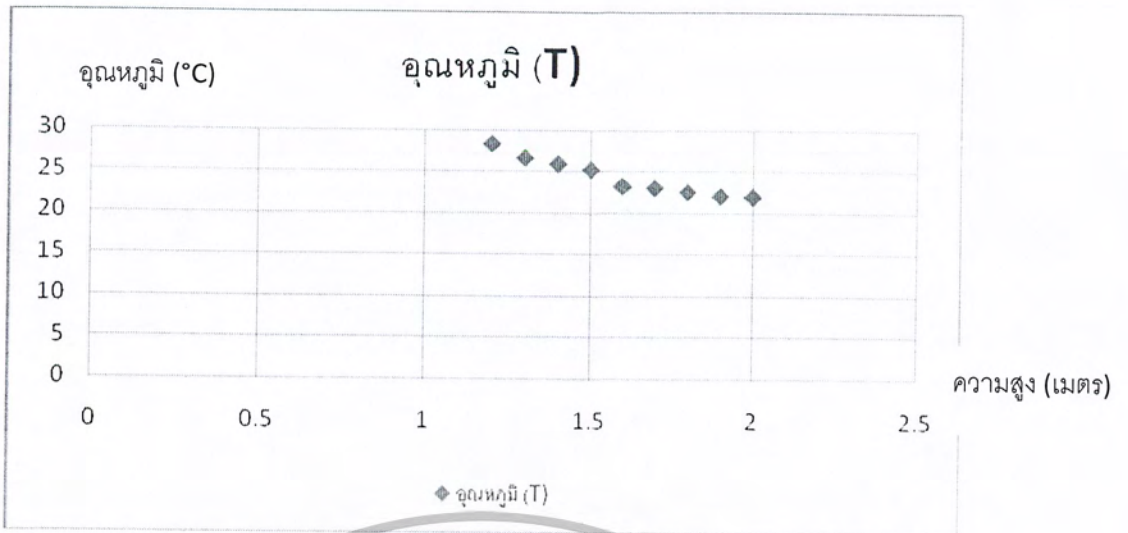


รูปที่ 5.9 แสดงกราฟผลการทดลองของอัตราการไหลของน้ำที่ระดับความสูงต่างๆ

ตารางที่ 5.2 แสดงผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ท่อซิลิโคน

ความสูง (H)	อุณหภูมิ (T)
1.2	28.4
1.3	26.7
1.4	26.0
1.5	25.3
1.6	23.3
1.7	23.1
1.8	22.6
1.9	22.1
2.0	22.0

จากค่าอุณหภูมิในตารางที่ 5.2 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.10

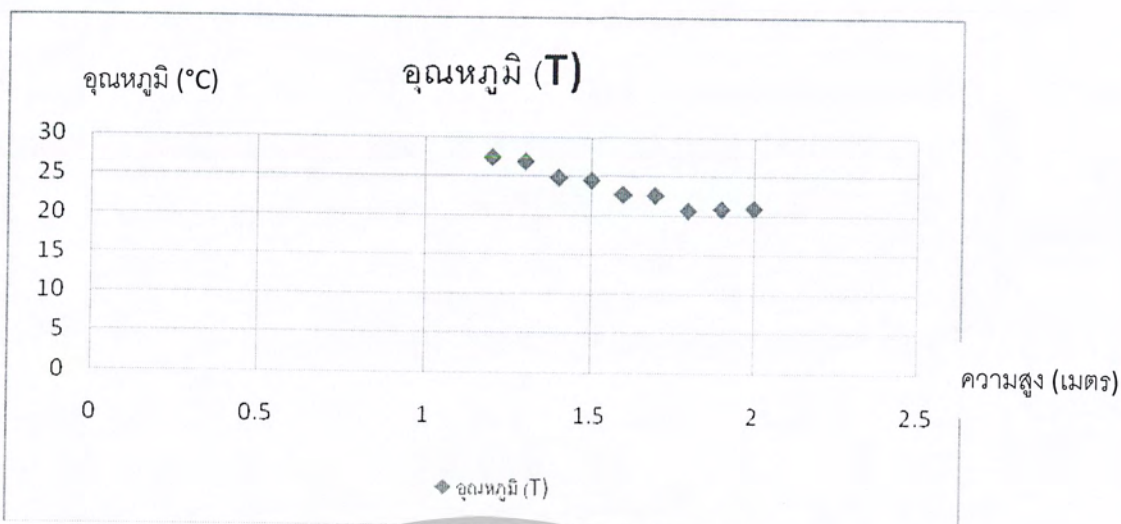


รูปที่ 5.10 แสดงกราฟผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ซิลิโคน

ตารางที่ 5.3 แสดงผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ท่อ PVC

ความสูง (H)	อุณหภูมิ (T)
1.2	27.4
1.3	26.9
1.4	24.9
1.5	24.6
1.6	22.8
1.7	22.7
1.8	20.8
1.9	21.0
2.0	21.1

จากค่าอุณหภูมิในตารางที่ 5.3 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.11



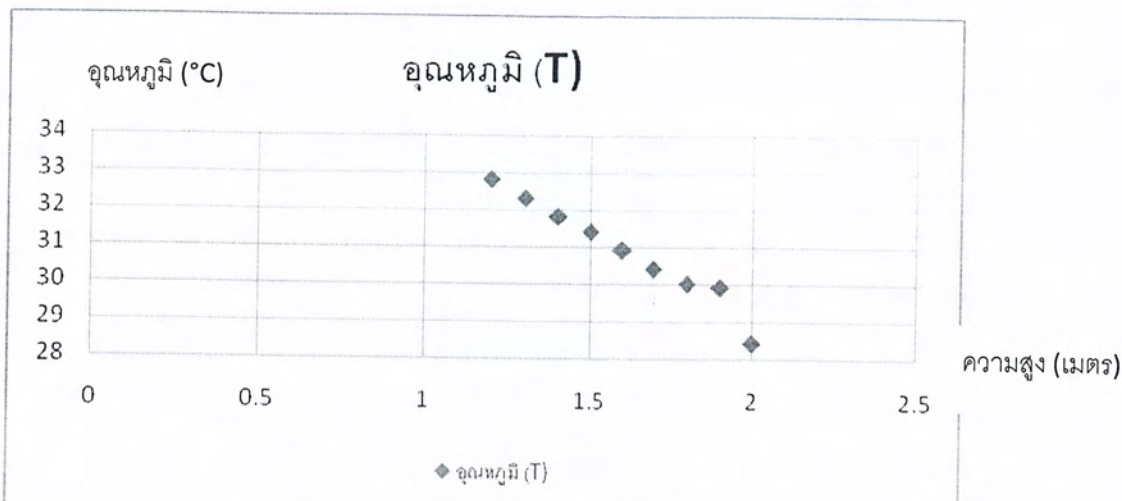
รูปที่ 5.11 แสดงกราฟผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ท่อ PVC

ตารางที่ 5.4 แสดงผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ช่องว่างซึ่ง

มีฟอยล์หุ้ม

ความสูง (H)	อุณหภูมิ (T)
1.2	32.8
1.3	32.3
1.4	31.8
1.5	31.4
1.6	30.9
1.7	30.4
1.8	30.0
1.9	29.9
2.0	28.4

จากค่าอุณหภูมิในตารางที่ 5.4 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.12



รูปที่ 5.12 กราฟผลการทดลองของการวัดอุณหภูมิของน้ำ โดยวัดอุณหภูมิบริเวณ ช่องว่างซึ่ง

มีพอยต์หุ้ม

ตารางที่ 5.5 แสดงผลการทดลองการวัดกำลังของปั้มน้ำ

ระยะสูงสุดของน้ำที่ปั้มน้ำสามารถปั้มขึ้นได้	1.20 m
--	--------

ตารางที่ 5.6 แสดงผลการทดลองการวัดอุณหภูมิด้วยเซนเซอร์

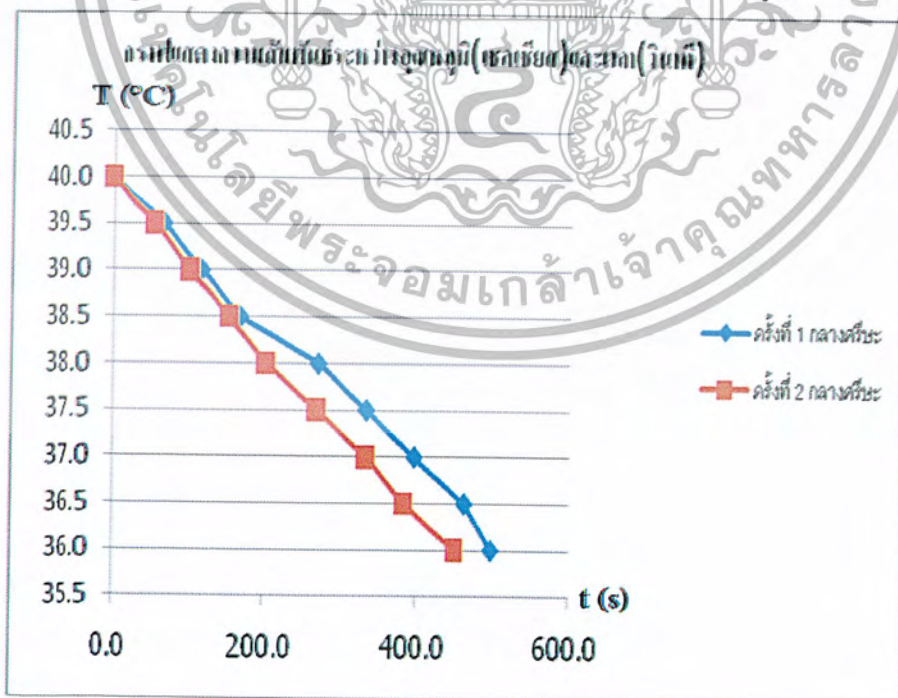
ชนิดของสารที่ใช้วัด	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Thermometer
ไอน้ำแข็ง	17.5	17.0	16.5	17.5	17.0
อุณหภูมิห้อง	31.5	31.5	31.5	31.5	31.5
น้ำที่อุณหภูมิห้อง	59.5	60.0	59.5	60.5	61.0
ไอน้ำร้อน	31.0	31.0	30.0	30.5	30.5

ตารางที่ 5.7 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา

โดย Sensor ตัวที่ 1 (บริเวณกลางศีรษะ)

อุณหภูมิ(เซลเซียส)	ครั้งที่ 1	ครั้งที่ 2
	เวลา(วินาที)	เวลา(วินาที)
40.0	0.0	0.00
39.5	66.0	54.00
39.0	115.0	102.0
38.5	167.0	153.0
38.0	270.0	202.0
37.5	333.0	269.0
37.0	396.0	334.0
36.5	462.0	384.0
36.0	497.0	451.0

จากค่าตารางที่ 5.7 นำมาสร้างเป็นรูปภาพดังแสดงในรูปที่ 5.13



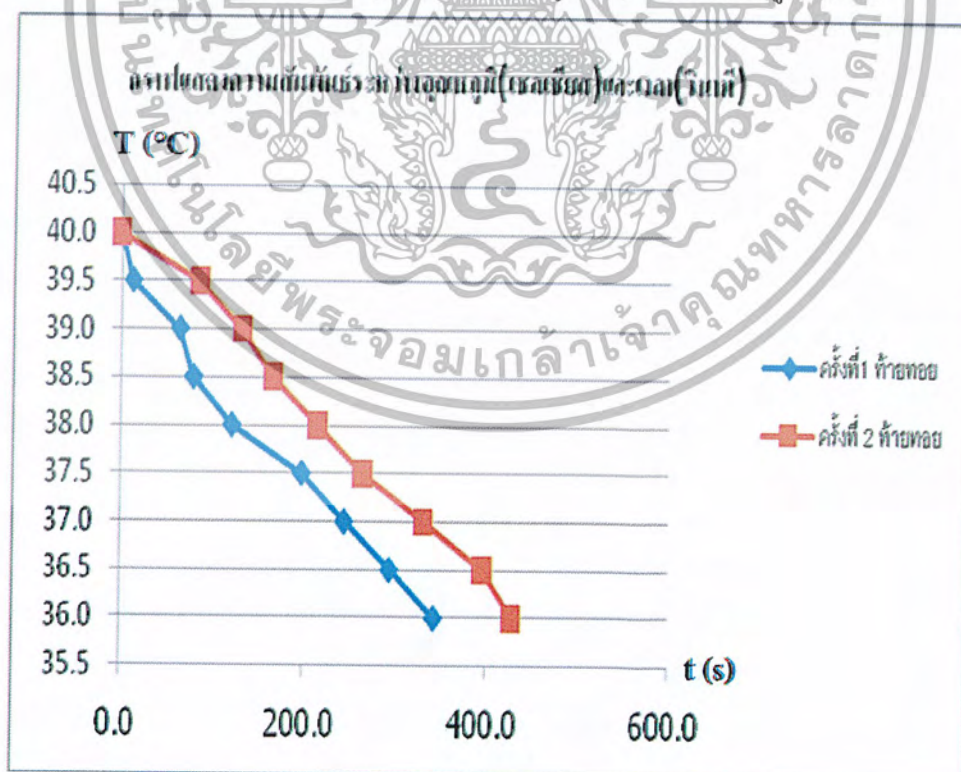
รูปที่ 5.13 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลาโดย Sensor ตัวที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน (บริเวณกลางศีรษะ) เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.8 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา
โดย Sensor ตัวที่ 2 (บริเวณท้ายทอย)

อุณหภูมิ(เซลเซียส)	ครั้งที่ 1	ครั้งที่ 2
	เวลา(วินาที)	เวลา(วินาที)
40.0	0.0	0.0
39.5	12.1	85.0
39.0	63.0	132.0
38.5	78.0	166.0
38.0	120.0	215.0
37.5	197.0	265.0
37.0	244.0	332.0
36.5	294.0	397.0
36.0	343.0	429.0

จากค่าในตารางที่ 5.8 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.14



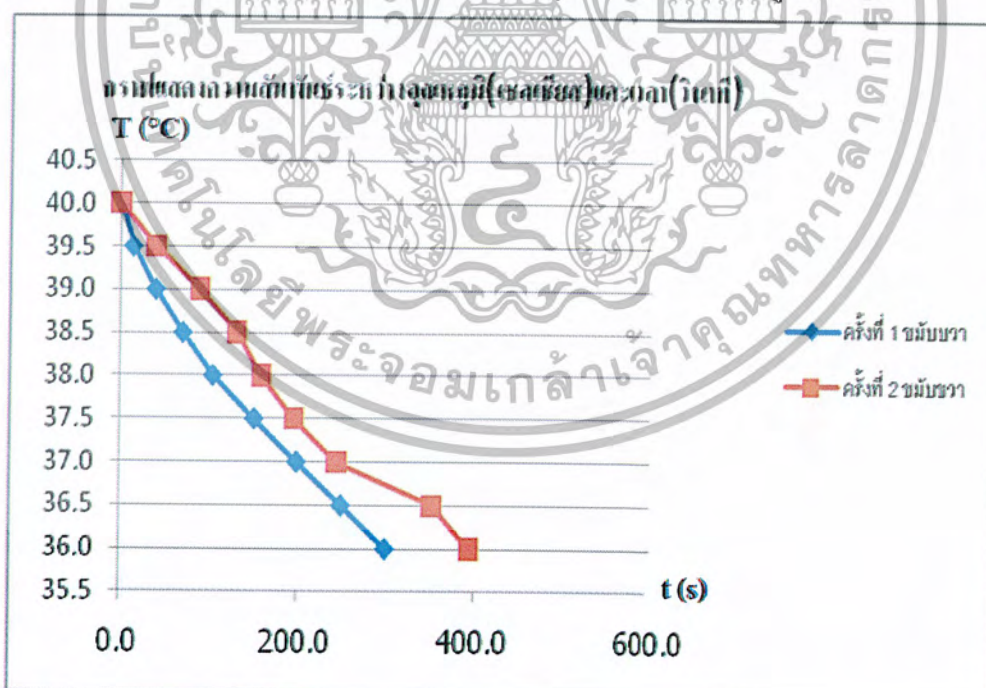
รูปที่ 5.14 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน (บริเวณท้ายทอย) เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.9 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 3 (บริเวณขมับขวา)

อุณหภูมิ(เซลเซียส)	ครั้งที่ 1 เวลา(วินาที)	ครั้งที่ 2 เวลา(วินาที)
40.0	0.0	0.0
39.5	13.1	40.2
39.0	39.4	90.0
38.5	70.0	132.0
38.0	104.0	160.0
37.5	151.0	197.0
37.0	199.0	246.0
36.5	249.0	353.0
36.0	299.0	395.0

จากค่าในตารางที่ 5.9 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.15

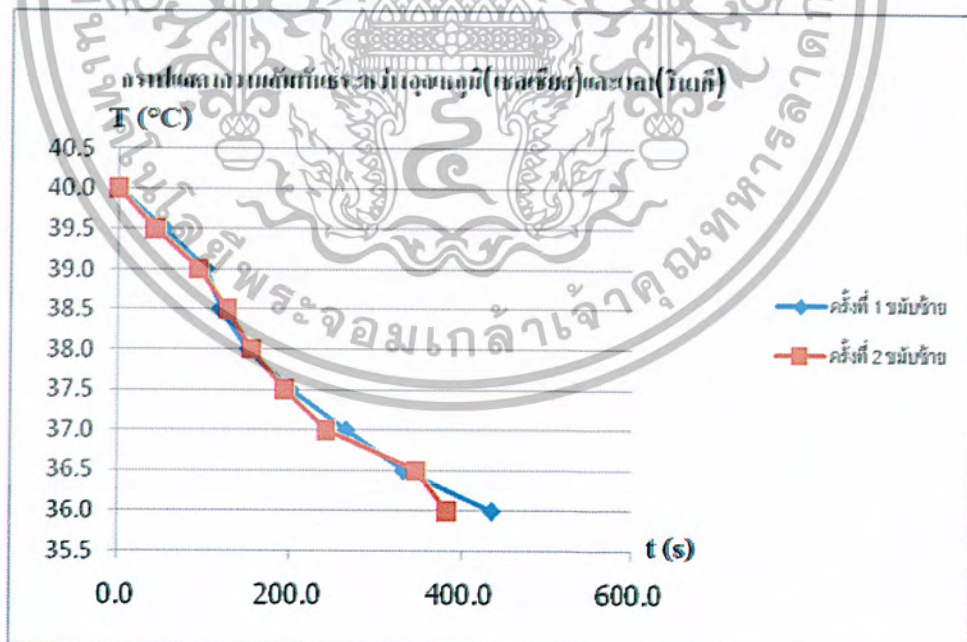


รูปที่ 5.15 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดย Sensor ตัวที่ 3 (บริเวณขมับขวา)

ตารางที่ 5.10 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา
โดย Sensor ตัวที่ 4 (บริเวณขมับซ้าย)

	ครั้งที่ 1	ครั้งที่ 2
อุณหภูมิ(เซลเซียส)	เวลา(วินาที)	เวลา(วินาที)
40.0	0.0	0.0
39.5	51.0	42.0
39.0	100.0	93.0
38.5	117.0	127.0
38.0	150.0	155.0
37.5	200.0	194.0
37.0	265.0	243.0
36.5	332.0	348.0
36.0	435.0	384.0

จากค่าในตารางที่ 5.10 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.16



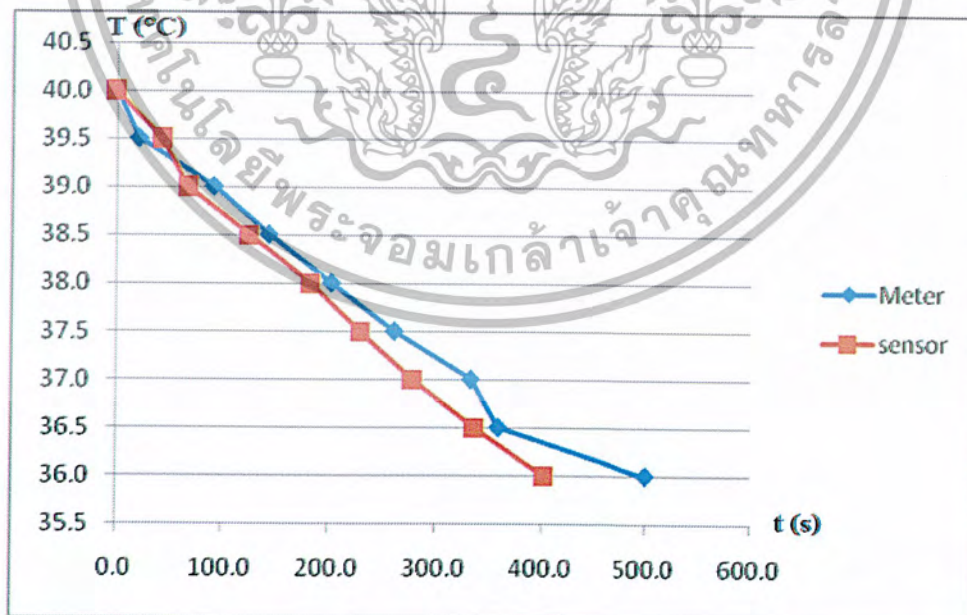
รูปที่ 5.16 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลาโดย Sensor ตัวที่ 4
(บริเวณขมับซ้าย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.11 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่าง
เทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น

อุณหภูมิ(เซลเซียส)	อ่านได้จาก Meter (FLUKE) (วินาที)	อ่านได้จาก SENSOR ที่ติดบน ศีรษะ (วินาที)
40.0	0.0	0
39.5	21.0	43
39.0	91.1	68
38.5	143.0	125
38.0	202.0	183
37.5	261.0	230
37.0	333.0	279
36.5	359.0	337
36.0	499.0	403

จากค่าในตารางที่ 5.11 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.17



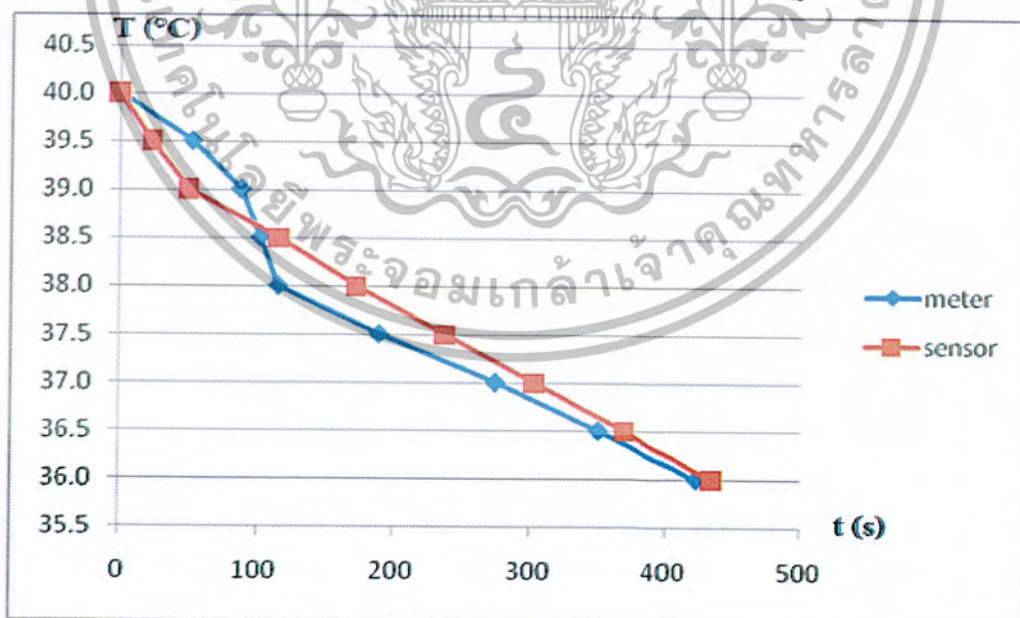
รูปที่ 5.17 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่าง
เทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.12 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =12 V (อุณหภูมิน้ำ 25 °C , อัตราการไหล 250 ml/min)

อุณหภูมิ(เซลเซียส)	อ่านได้จาก Meter (FLUKE) (วินาที)	อ่านได้จาก SENSOR ที่ติดบนสิริษะ (วินาที)
40.0	0	0
39.5	53	24
39.0	88	51
38.5	102	116
38.0	115	173
37.5	189	238
37.0	274	303
36.5	349	369
36.0	423	435

จากค่าในตารางที่ 5.12 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.18



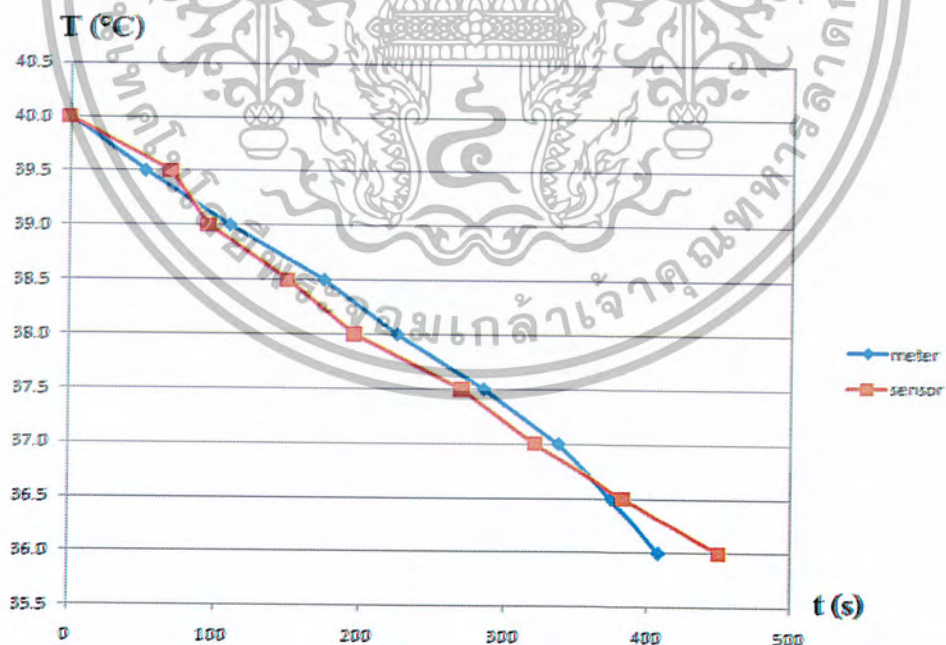
รูปที่ 5.18 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =12 V (อุณหภูมิน้ำ 25 °C , อัตราการไหล 250 ml/min)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.13 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =9 V
(อุณหภูมิน้ำ 25 °C , อัตราการไหล 230 ml/min)

อุณหภูมิ(เซลเซียส)	อ่านได้จาก Meter (FLUKE) (วินาที)	อ่านได้จาก SENSOR ที่ติดบนศิรชะ (วินาที)
40.0	0	0
39.5	51	69
39.0	109	95
38.5	174	150
38.0	225	196
37.5	285	271
37.0	336	321
36.5	374	383
36.0	407	450

จากค่าในตารางที่ 5.13 นำมาสร้างเป็นรูปภาพดังแสดงในรูปที่ 5.19



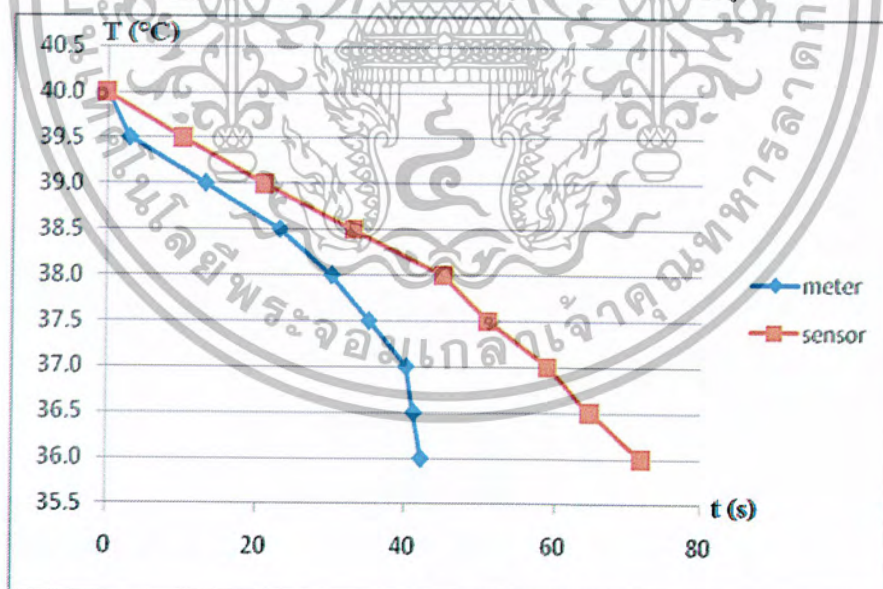
รูปที่ 5.19 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =9 V
(อุณหภูมิน้ำ 25 °C , อัตราการไหล 230 ml/min)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.14 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =12 V (อุณหภูมิน้ำ 5 °C , อัตราการไหล 250 ml/min)

อุณหภูมิ(เซลเซียส)	อ่านได้จาก Meter (FLUKE) (วินาที)	อ่านได้จาก SENSOR ที่ติดบน ศีรษะ (วินาที)
40.0	0	0
39.5	3	10
39.0	13	21
38.5	23	33
38.0	30	45
37.5	35	51
37.0	40	59
36.5	41	65
36.0	42	72

จากค่าในตารางที่ 5.14 นำมาสร้างเป็นรูปภาพดังแสดงในรูปที่ 5.20



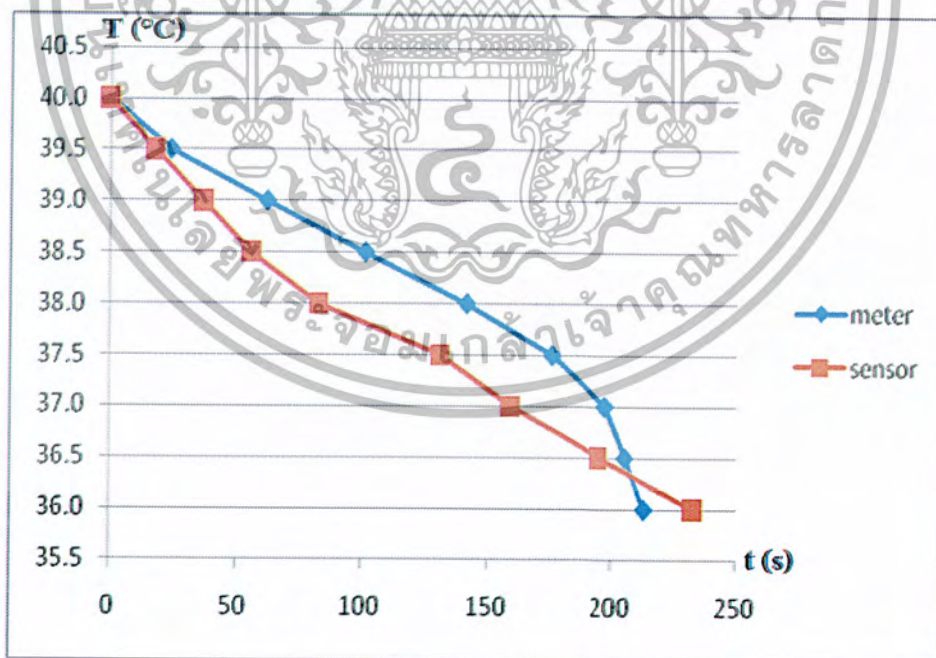
รูปที่ 5.20 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั้มน้ำด้วย DC =12 V (อุณหภูมิน้ำ 5 °C , อัตราการไหล 250 ml/min)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.15 แสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั๊มน้ำด้วย DC =9 V
(อุณหภูมิน้ำ 5 °C , อัตราการไหล 230 ml/min)

อุณหภูมิ(เซลเซียส)	อ่านได้จาก Meter (FLUKE) (วินาที)	อ่านได้จาก SENSOR ที่ติดบนศีรษะ (วินาที)
40.0	0	0
39.5	24	18
39.0	62	37
38.5	101	56
38.0	141	83
37.5	175	131
37.0	197	159
36.5	205	195
36.0	213	233

จากค่าในตารางที่ 5.15 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 5.21



รูปที่ 5.21 กราฟแสดงผลการทดลองของอัตราการเปลี่ยนแปลงอุณหภูมิต่อเวลา โดยเปรียบเทียบระหว่างเทอร์มอมิเตอร์จริงกับเทอร์มอมิเตอร์ที่สร้างขึ้น โดยจ่ายไฟให้ปั๊มน้ำด้วย DC =9 V
(อุณหภูมิน้ำ 5 °C , อัตราการไหล 230 ml/min)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.16 แสดงผลการทดลองของการป้อนค่า PWM $V_{oH} = 5\text{ V}$, $V_{oL} = 0\text{ V}$ ที่ Duty Cycle 99.5% , 75% , 50% , 25% โดยใช้ความถี่ 10 KHz 5 KHz 50Hz

Duty Cycle	อัตราการไหล (ml/min)		
	10 KHz	5 KHz	50 Hz
99.50%	1350	1355	1335
75%	1320	1180	1200
50%	1035	620	960
25%	425	200	630

ตารางที่ 5.17 แสดงผลการทดลองค่าอัตราการไหล ที่ Duty Cycle 99.5% , 75% , 50% , 25% โดยใช้ความถี่ 1.2 KHz (จาก PIC 16F877A)

Duty Cycle	อัตราการไหล (ml/min)
99.50%	1305
75%	1020
50%	840
25%	450

บทที่ 6

สรุปและวิจารณ์

6.1 สรุปผลการทดลอง

การทดลองที่ 1

การทดลองวัดอัตราการไหลของน้ำ ขณะที่ไม่มีมอเตอร์ และไม่มีมอเตอร์

จากการทดลองพบว่า การนำปั้มน้ำมาไว้ตรงบริเวณที่น้ำไหลออกจากหมวกนั้นทำให้อัตราการไหลของน้ำที่ไหลออกมาในเวลา 1 นาที มีประมาณมากกว่าที่ไหลออกจากหมวกโดยไม่มีปั้มน้ำมาช่วยในการดึงน้ำออกจากหมวก

การทดลองที่ 2

การทดลองวัดอุณหภูมิบริเวณต่างๆของหมวกลดไข้

จากการวัดอุณหภูมิของน้ำขณะผ่านหมวกพบว่า พบว่า เมื่อยกระดับของขวดน้ำให้สูงกว่าระดับหมวก น้ำก็จะไหลผ่านหมวกเรื่อยๆ พอวัดอุณหภูมิที่หมวกพบว่าอุณหภูมิลดลง และพอยกขวดน้ำสูงขึ้นเรื่อยๆ จะพบว่าอุณหภูมิลดต่ำลงเรื่อยๆ ซึ่งการวัดอุณหภูมิที่บริเวณสายยางและบริเวณท่อที่มีการลดของอุณหภูมิดีมากกว่าการวัดที่บริเวณที่มีกระดาษพอยด์หุ้มอยู่ แต่อุณหภูมิที่ได้ก็เป็นอุณหภูมิต่ำกว่าอุณหภูมิของร่างกาย ดังนั้นอุณหภูมิน้ำที่ผ่านหมวกก็เป็นตัวช่วยในการระบายความร้อนของร่างกายได้

การทดลองที่ 3

การทดลอง วัดกำลังของปั้มน้ำ

พบว่าปั้มน้ำมีกำลังในการปั้มน้ำได้สูงสุดที่ตำแหน่ง 1.2 m

การทดลองที่ 4

การทดลองวัดโดยใช้เซนเซอร์

จากการทดลองโดยนำเซนเซอร์มาวัดค่าอุณหภูมิของสิ่งต่างที่กำหนดนั้นคือ ใช้น้ำแข็ง อุณหภูมิห้อง น้ำธรรมดาที่อุณหภูมิห้อง น้ำอุ่น โดยวัดเทียบกับเทอร์โมมิเตอร์มาตรฐานจะพบว่า ค่าที่วัดได้จากเซนเซอร์ของวงจรเมื่อเทียบกับเทอร์โมมิเตอร์มาตรฐานมีค่าต่างกันไม่มากนักอาจเนื่องมาจากสาเหตุบางประการ สามารถสรุปได้ว่าเมื่อวัดอุณหภูมิใช้น้ำแข็งจะพบว่าเซนเซอร์ตัวที่ 1 และ 4 มีค่าใกล้เคียงค่าจริงมากที่สุด เมื่อวัดอุณหภูมิห้องจะพบว่าเซนเซอร์ทุกตัววัดค่าอุณหภูมิได้ค่าเท่ากันหมดซึ่งเท่ากับค่าจริงคือ 31.5 องศาเซลเซียส เมื่อวัดอุณหภูมิน้ำที่อุณหภูมิห้องจะพบว่าเซนเซอร์ตัวที่ 3 มีค่าเท่ากับค่าจริง เมื่อวัดอุณหภูมิน้ำอุ่นจะพบว่าเซนเซอร์ตัวที่ 1 มีค่าใกล้เคียงค่าจริงมากที่สุด

การทดลองที่ 5

การทดลองการวัดอุณหภูมิของเซนเซอร์ DS 1820

1. Sensor มีการตอบสนองต่ออุณหภูมิได้ไว
2. ที่กลางศีรษะจะมีอุณหภูมิที่ร้อนมากที่สุดเพราะถูกลมร้อนเป่าเข้าไปโดยตรง

การทดลองที่ 6

การทดลองการวัดอุณหภูมิของเซนเซอร์ DS 1820 กับ เทอร์โมมิเตอร์จริง (Meter)

1. เมื่อวัดค่าจาก Sensor และ บีมน์น้ำ พบว่ามีการคลาดเคลื่อนไปเล็กน้อย (เนื่องจาก Sensor ที่ใช้วัดมีการ delay ในการอ่านค่า)
2. อุณหภูมิมีการลดลงได้เองอย่างต่อเนื่อง (แต่เป็นไปอย่างช้าๆ)

การทดลองที่ 7

การทดลองการวัดอุณหภูมิของเซนเซอร์ DS 1820 กับ เทอร์โมมิเตอร์จริง (Meter) โดยจ่ายไฟ 12Vdc และ 9 Vdc ที่ T=25 °C

1. ที่ DC Voltage ต่างกันไป ก็จะทำให้อัตราการไหลมีค่าต่างกัน กล่าวคือ เมื่อ volt ลดลง อัตราการไหลก็จะลดลงด้วย

2. จากกราฟ เมื่อ Flow น้ำ ที่มีอัตราการไหลมากก็จะทำให้อุณหภูมิลดลงเร็วมากขึ้น ($12V > 9V > 6V$)

การทดลองที่ 8

การทดลองการวัดอุณหภูมิของเซนเซอร์ DS 18B20 กับ เทอร์มอมิเตอร์จริง (Meter) โดยจ่ายไฟ 12Vdc และ 9 Vdc ที่ $T = 5\text{ }^{\circ}\text{C}$

จากการทดลองพบว่า อุณหภูมิของน้ำเมื่อยิ่งต่ำลง ($= 5\text{ }^{\circ}\text{C}$) ก็จะทำให้อุณหภูมิที่บริเวณศีรษะของหุ่นลดลงเร็วมากขึ้น

การทดลองที่ 9

1. เมื่อให้ปั้มน้ำทำงานเต็มที่ ให้อัตราการไหล = 2100 ml/min ได้ระดับน้ำสูง มากกว่า 3 เมตร
2. เมื่อใช้ปั้มน้ำ 2 ตัว คือ ปั้มน้ำเข้าและสูบน้ำออก พบว่า อัตราการไหลลดลง เนื่องจากปั้มน้ำตัวที่ 2 (ปั้มน้ำสูบน้ำออก) กลายเป็นตัวที่ไปขวางทางน้ำไหลออก

การทดลองที่ 10

การทดลองการใช้งานของหมวก

1. พบว่าศีรษะรู้สึกเย็นมาก
2. มีไอน้ำมาเกาะบริเวณหมวก
3. เกิดเสียงดังเนื่องจากน้ำที่ไหลภายในหมวก (เสียงคล้ายฟองอากาศของน้ำ)

การทดลองที่ 11

การทดลองค่าอัตราการไหลของน้ำที่ค่า PWM ค่าต่างๆ

จากการทดลองป้อนค่า PWM ด้วยค่า Duty Cycle ที่เพิ่มขึ้นพบว่า อัตราการไหลของน้ำก็เพิ่มขึ้นด้วย

6.2 วิจารณ์ผลการทดลอง

ผลการทดลองอาจมีความคลาดเคลื่อนเล็กน้อยเนื่องจากสาเหตุ

1. ในขณะนี้ยังไม่สามารถควบคุมอุณหภูมิของน้ำที่ปล่อยไหลผ่านหมวกได้
2. ในการวัดอุณหภูมิที่หมวกอาจไม่คงที่เนื่องจากอุณหภูมิภายนอกไม่คงที่
3. เกิดการควบแน่นของน้ำบริเวณรอบๆหมวก
4. บรรยากาศภายในห้องมีพัดลมเป่าตลอดเวลาทำให้ค่าที่ได้จากการวัดอุณหภูมิในวัตถุที่ร้อนและเย็นมีค่าไม่คงที่
5. ความลึกในการวัดค่าอุณหภูมิของวัตถุของเหลวที่มีความร้อนมีค่าไม่เท่ากันระหว่างเทอร์โมมิเตอร์มาตรฐานและเซนเซอร์เนื่องจากอุณหภูมิที่ผิวหน้าของของเหลวมีค่าไม่เท่ากับอุณหภูมิภายในของเหลวเมื่อความลึกในการวัดค่าไม่เท่ากันจึงทำให้ค่าอุณหภูมิที่วัดได้มีความคลาดเคลื่อนผิดจากค่าจริงเล็กน้อย
6. ตำแหน่งในการวัดอุณหภูมิของเซนเซอร์และเทอร์โมมิเตอร์มาตรฐานไม่ใช่ตำแหน่งเดียวกันจึงทำให้ค่าอุณหภูมิที่วัดได้มีความคลาดเคลื่อนผิดจากค่าจริงเล็กน้อย
7. การวัดอุณหภูมิน้ำอุ่นของเซนเซอร์และเทอร์โมมิเตอร์มาตรฐานจะได้ค่าที่นำมาเปรียบเทียบไม่ใช่ค่าจริงถ้าไม่ได้วัดในเวลาเดียวกันเนื่องจากอุณหภูมิของน้ำอุ่นจะเปลี่ยนแปลงตามเวลาถ้าเวลาที่ยิ่งผ่านไปนานอุณหภูมิจะยิ่งลดลง
8. ค่าจากการทดลองอาจคลาดเคลื่อนเนื่องจากผู้ทำการวัดขาดความชำนาญในการใช้อุปกรณ์
9. ค่าจากการทดลองอาจมีความคลาดเคลื่อน จาก Delay ของ Sensor ที่ทำการวัด (เมื่อเทียบกับมิเตอร์มาตรฐาน)
10. หากวงจร Supply ไม่สามารถจ่ายกระแสได้เพียงพอจะทำให้ Voltage drop ลง
11. การทดลองจะต้องทำการทดลองกับสรีระของหุ่นจำลอง เนื่องจากเราไม่สามารถวัดกับคนจริงๆได้ เนื่องจากอุปกรณ์ยังอยู่ในช่วงพัฒนาและขณะทำการวิจัยไม่สามารถหาคนไข้ที่มีอุณหภูมิของร่างกายสูง (มีไข้) มาทดลองได้จริง ซึ่งอาจทำให้ค่าที่ได้มีความผิดพลาดเกิดขึ้น เพราะสรีระของหุ่นจำลองมีวัสดุที่ต่างจากสรีระของมนุษย์

6.3 ข้อผิดพลาดที่เกิดขึ้นระหว่างการทดลอง

1. Power Supply จ่ายกระแสไม่เพียงพอ
2. จอ LCD มีปัญหา ทำให้ส่วนอื่นของวงจรได้รับความเสียหายด้วย
3. GND ที่ใช้ในวงจรไม่ได้แยกกัน ซึ่งทำให้วงจรได้รับความเสียหายได้
4. ข้อต่อของสายขงอาจหลุดออกมาระหว่างทำการทดลองทำให้พื้นที่ที่ทำการทดลองเปียกและอุปกรณ์อื่นๆได้รับความเสียหายได้



เอกสารอ้างอิง

- [1] <http://www.oknation.net>
- [2] <http://www.doctor.or.th/>
- [3] <http://www.bangkokhealth.com/>
- [4] <http://www.bloggang.com/>
- [5] <http://www.hopeofsamutsakhon.com/>
- [6] <http://topicstock.pantip.com/wahkor/topicstock/>
- [7] <http://www.vcharkarn.com/vcafe/>
- [8] <http://www.thaimcu.com/>
- [9] <http://www.thaiembedded.com/blog/?tag=ds1820>





ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา **xiii** และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FEATURES

- Unique 1-wire interface requires only one port pin for communication
- Each device has a unique 64-bit serial code stored in an on-board ROM
- Multi-drop capability simplifies distributed temperature sensing applications
- Requires no external components
- Can be powered from data line. Power supply range is 3.0V to 5.5V
- Measures temperatures from -55°C to $+125^{\circ}\text{C}$ (-67°F to $+257^{\circ}\text{F}$)
- $\pm 0.5^{\circ}\text{C}$ accuracy from -10°C to $+85^{\circ}\text{C}$
- 9-bit thermometer resolution
- Converts temperature in 750 ms (max.)
- User-definable nonvolatile 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



(BOTTOM VIEW)

TO-92
(DS18S20)



8-pin 150-mil SOIC
(DS18S20Z)

PIN DESCRIPTION

- GND - Ground
 DQ - Data In/Out
 V_{DD} - Power Supply Voltage
 NC - No Connect

DESCRIPTION

The DS18S20 Digital Thermometer provides 9-bit centigrade temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18S20 communicates over a 1-wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. It has an operating temperature range of -55°C to $+125^{\circ}\text{C}$ and is accurate to $\pm 0.5^{\circ}\text{C}$ over the range of -10°C to $+85^{\circ}\text{C}$. In addition, the DS18S20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply.

Each DS18S20 has a unique 64-bit serial code, which allows multiple DS18S20s to function on the same 1-wire bus; thus, it is simple to use one microprocessor to control many DS18S20s distributed over a large area. Applications that can benefit from this feature include HVAC environmental controls, temperature monitoring systems inside buildings, equipment or machinery, and process monitoring and control systems.

DETAILED PIN DESCRIPTIONS Table 1

8-PIN SOIC*	TO-92	SYMBOL	DESCRIPTION
5	1	GND	Ground.
4	2	DQ	Data Input/Output pin. Open-drain 1-wire interface pin. Also provides power to the device when used in parasite power mode (see "Parasite Power" section.)
3	3	V _{DD}	Optional V_{DD} pin. V _{DD} must be grounded for operation in parasite power mode.

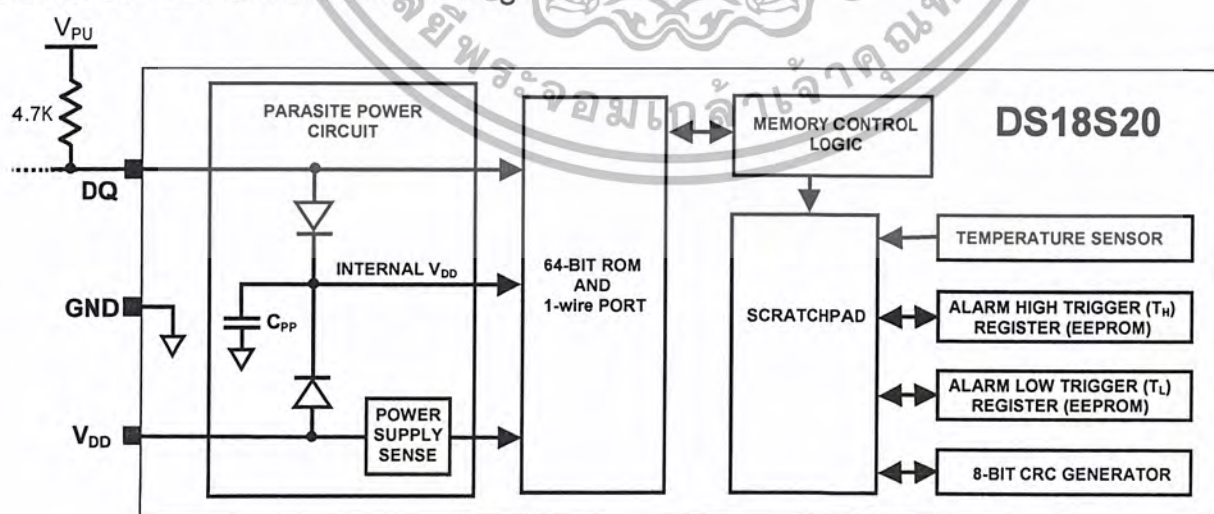
*All pins not specified in this table are "No Connect" pins.

OVERVIEW

Figure 1 shows a block diagram of the DS18S20, and pin descriptions are given in Table 1. The 64-bit ROM stores the device's unique serial code. The scratchpad memory contains the 2-byte temperature register that stores the digital output from the temperature sensor. In addition, the scratchpad provides access to the 1-byte upper and lower alarm trigger registers (T_H and T_L). The T_H and T_L registers are nonvolatile (EEPROM), so they will retain data when the device is powered down.

The DS18S20 uses Dallas' exclusive 1-wire bus protocol that implements bus communication using one control signal. The control line requires a weak pullup resistor since all devices are linked to the bus via a 3-state or open-drain port (the DQ pin in the case of the DS18S20). In this bus system, the microprocessor (the master device) identifies and addresses devices on the bus using each device's unique 64-bit code. Because each device has a unique code, the number of devices that can be addressed on one bus is virtually unlimited. The 1-wire bus protocol, including detailed explanations of the commands and "time slots," is covered in the 1-WIRE BUS SYSTEM section of this datasheet.

Another feature of the DS18S20 is the ability to operate without an external power supply. Power is instead supplied through the 1-wire pullup resistor via the DQ pin when the bus is high. The high bus signal also charges an internal capacitor (C_{PP}), which then supplies power to the device when the bus is low. This method of deriving power from the 1-wire bus is referred to as "parasite power." As an alternative, the DS18S20 may also be powered by an external supply on V_{DD}.

DS18S20 BLOCK DIAGRAM Figure 1

OPERATION – MEASURING TEMPERATURE

The core functionality of the DS18S20 is its direct-to-digital temperature sensor. The temperature sensor output has 9-bit resolution, which corresponds to 0.5°C steps. The DS18S20 powers-up in a low-power idle state; to initiate a temperature measurement and A-to-D conversion, the master must issue a Convert T [44h] command. Following the conversion, the resulting thermal data is stored in the 2-byte temperature register in the scratchpad memory and the DS18S20 returns to its idle state. If the DS18S20 is powered by an external supply, the master can issue “read time slots” (see the 1-WIRE BUS SYSTEM section) after the Convert T command and the DS18S20 will respond by transmitting 0 while the temperature conversion is in progress and 1 when the conversion is done. If the DS18S20 is powered with parasite power, this notification technique cannot be used since the bus must be pulled high by a strong pullup during the entire temperature conversion. The bus requirements for parasite power are explained in detail in the POWERING THE DS18S20 section of this datasheet.

The DS18S20 output data is calibrated in degrees centigrade; for Fahrenheit applications, a lookup table or conversion routine must be used. The temperature data is stored as a 16-bit sign-extended two’s complement number in the temperature register (see Figure 2). The sign bits (S) indicate if the temperature is positive or negative: for positive numbers $S = 0$ and for negative numbers $S = 1$. Table 2 gives examples of digital output data and the corresponding temperature reading.

Resolutions greater than 9 bits can be calculated using the data from the temperature, COUNT REMAIN and COUNT PER °C registers in the scratchpad. Note that the COUNT PER °C register is hard-wired to 16 (10h). After reading the scratchpad, the TEMP_READ value is obtained by truncating the 0.5°C bit (bit 0) from the temperature data (see Figure 2). The extended resolution temperature can then be calculated using the following equation:

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

Additional information about high-resolution temperature calculations can be found in Application Note 105: “High Resolution Temperature Measurement with Dallas Direct-to-Digital Temperature Sensors”.

TEMPERATURE REGISTER FORMAT Figure 2

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
LS Byte	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹
	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
MS Byte	S	S	S	S	S	S	S	S

TEMPERATURE/DATA RELATIONSHIP Table 2

TEMPERATURE	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+85.0°C*	0000 0000 1010 1010	00AAh
+25.0°C	0000 0000 0011 0010	0032h
+0.5°C	0000 0000 0000 0001	0001h
0°C	0000 0000 0000 0000	0000h
-0.5°C	1111 1111 1111 1111	FFFFh
-25.0°C	1111 1111 1100 1110	FFCEh
-55.0°C	1111 1111 1001 0010	FF92h

*The power-on reset value of the temperature register is +85°C

OPERATION – ALARM SIGNALING

After the DS18S20 performs a temperature conversion, the temperature value is compared to the user-defined two's complement alarm trigger values stored in the 1-byte T_H and T_L registers (see Figure 3). The sign bit (S) indicates if the value is positive or negative: for positive numbers $S = 0$ and for negative numbers $S = 1$. The T_H and T_L registers are nonvolatile (EEPROM) so they will retain data when the device is powered down. T_H and T_L can be accessed through bytes 2 and 3 of the scratchpad as explained in the MEMORY section of this datasheet.

T_H AND T_L REGISTER FORMAT Figure 3

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
S	2^6	2^5	2^5	2^5	2^2	2^1	2^0

Only bits 8 through 1 of the temperature register are used in the T_H and T_L comparison since T_H and T_L are 8-bit registers. If the result of a temperature measurement is higher than T_H or lower than T_L , an alarm condition exists and an alarm flag is set inside the DS18S20. This flag is updated after every temperature measurement; therefore, if the alarm condition goes away, the flag will be turned off after the next temperature conversion.

The master device can check the alarm flag status of all DS18S20s on the bus by issuing an Alarm Search [ECh] command. Any DS18S20s with a set alarm flag will respond to the command, so the master can determine exactly which DS18S20s have experienced an alarm condition. If an alarm condition exists and the T_H or T_L settings have changed, another temperature conversion should be done to validate the alarm condition.

POWERING THE DS18S20

The DS18S20 can be powered by an external supply on the V_{DD} pin, or it can operate in “parasite power” mode, which allows the DS18S20 to function without a local external supply. Parasite power is very useful for applications that require remote temperature sensing or that are very space constrained. Figure 1 shows the DS18S20's parasite-power control circuitry, which “steals” power from the 1-wire bus via the DQ pin when the bus is high. The stolen charge powers the DS18S20 while the bus is high, and some of the charge is stored on the parasite power capacitor (C_{PP}) to provide power when the bus is low. When the DS18S20 is used in parasite power mode, the V_{DD} pin must be connected to ground.

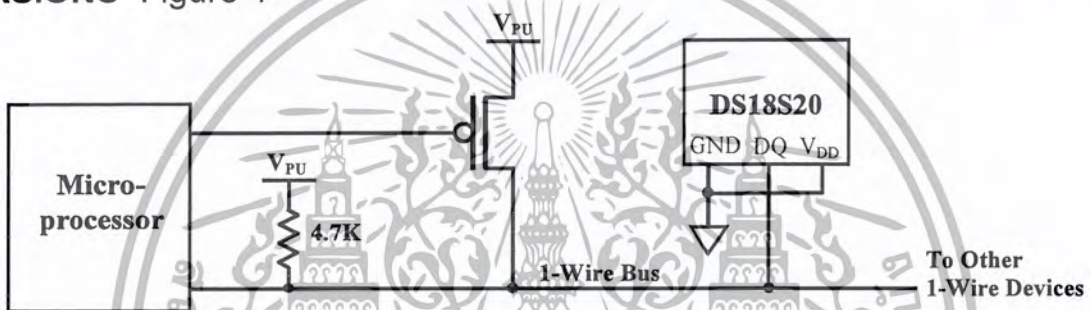
In parasite power mode, the 1-wire bus and C_{PP} can provide sufficient current to the DS18S20 for most operations as long as the specified timing and voltage requirements are met (refer to the DC ELECTRICAL CHARACTERISTICS and the AC ELECTRICAL CHARACTERISTICS sections of this data sheet). However, when the DS18S20 is performing temperature conversions or copying data from the scratchpad memory to EEPROM, the operating current can be as high as 1.5 mA. This current can cause an unacceptable voltage drop across the weak 1-wire pullup resistor and is more current than can be supplied by C_{PP} . To assure that the DS18S20 has sufficient supply current, it is necessary to provide a strong pullup on the 1-wire bus whenever temperature conversions are taking place or data is being copied from the scratchpad to EEPROM. This can be accomplished by using a MOSFET to pull the bus directly to the rail as shown in Figure 4. The 1-wire bus must be switched to the strong pullup within 10 μ s (max) after a Convert T [44h] or Copy Scratchpad [48h] command is issued, and the bus must be held high by the pullup for the duration of the conversion (t_{conv}) or data transfer ($t_{wr} = 10$ ms). No other activity can take place on the 1-wire bus while the pullup is enabled.

The DS18S20 can also be powered by the conventional method of connecting an external power supply to the V_{DD} pin, as shown in Figure 5. The advantage of this method is that the MOSFET pullup is not required, and the 1-wire bus is free to carry other traffic during the temperature conversion time.

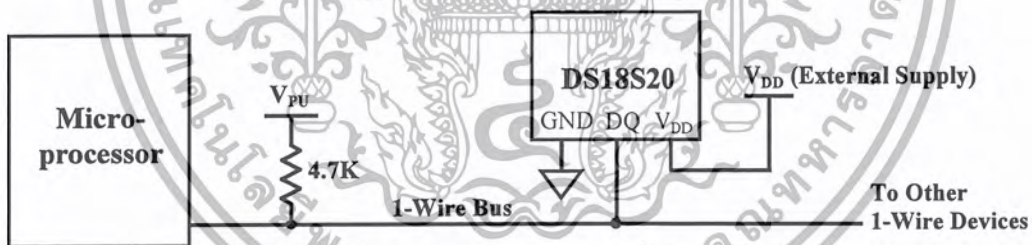
The use of parasite power is not recommended for temperatures above 100°C since the DS18S20 may not be able to sustain communications due to the higher leakage currents that can exist at these temperatures. For applications in which such temperatures are likely, it is strongly recommended that the DS18S20 be powered by an external power supply.

In some situations the bus master may not know whether the DS18S20s on the bus are parasite powered or powered by external supplies. The master needs this information to determine if the strong bus pullup should be used during temperature conversions. To get this information, the master can issue a Skip ROM [CCh] command followed by a Read Power Supply [B4h] command followed by a “read time slot”. During the read time slot, parasite powered DS18S20s will pull the bus low, and externally powered DS18S20s will let the bus remain high. If the bus is pulled low, the master knows that it must supply the strong pullup on the 1-wire bus during temperature conversions.

SUPPLYING THE PARASITE-POWERED DS18S20 DURING TEMPERATURE CONVERSIONS Figure 4



POWERING THE DS18S20 WITH AN EXTERNAL SUPPLY Figure 5



64-BIT LASERED ROM CODE

Each DS18S20 contains a unique 64-bit code (see Figure 6) stored in ROM. The least significant 8 bits of the ROM code contain the DS18S20’s 1-wire family code: 10h. The next 48 bits contain a unique serial number. The most significant 8 bits contain a cyclic redundancy check (CRC) byte that is calculated from the first 56 bits of the ROM code. A detailed explanation of the CRC bits is provided in the CRC GENERATION section. The 64-bit ROM code and associated ROM function control logic allow the DS18S20 to operate as a 1-wire device using the protocol detailed in the 1-WIRE BUS SYSTEM section of this datasheet.

64-BIT LASERED ROM CODE Figure 6

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

MEMORY

The DS18S20's memory is organized as shown in Figure 7. The memory consists of an SRAM scratchpad with nonvolatile EEPROM storage for the high and low alarm trigger registers (T_H and T_L). Note that if the DS18S20 alarm function is not used, the T_H and T_L registers can serve as general-purpose memory. All memory commands are described in detail in the DS18S20 FUNCTION COMMANDS section.

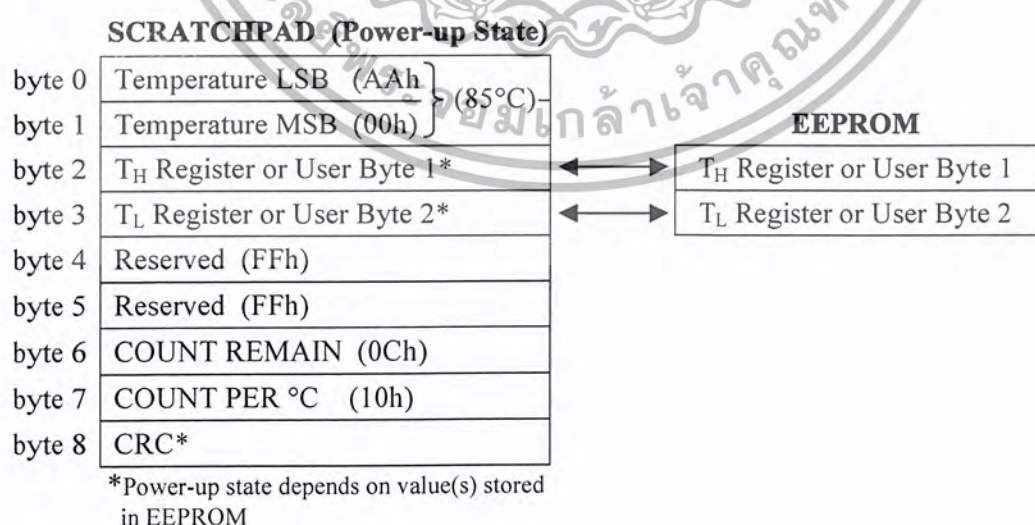
Byte 0 and byte 1 of the scratchpad contain the LSB and the MSB of the temperature register, respectively. These bytes are read-only. Bytes 2 and 3 provide access to T_H and T_L registers. Bytes 4 and 5 are reserved for internal use by the device and cannot be overwritten; these bytes will return all 1s when read. Bytes 6 and 7 contain the COUNT REMAIN and COUNT PER °C registers, which can be used to calculate extended resolution results as explained in the OPERATION – MEASURING TEMPERATURE section.

Byte 8 of the scratchpad is read-only and contains the cyclic redundancy check (CRC) code for bytes 0 through 7 of the scratchpad. The DS18S20 generates this CRC using the method described in the CRC GENERATION section.

Data is written to bytes 2 and 3 of the scratchpad using the Write Scratchpad [4Eh] command; the data must be transmitted to the DS18S20 starting with the least significant bit of byte 2. To verify data integrity, the scratchpad can be read (using the Read Scratchpad [BEh] command) after the data is written. When reading the scratchpad, data is transferred over the 1-wire bus starting with the least significant bit of byte 0. To transfer the T_H and T_L data from the scratchpad to EEPROM, the master must issue the Copy Scratchpad [48h] command.

Data in the EEPROM registers is retained when the device is powered down; at power-up the EEPROM data is reloaded into the corresponding scratchpad locations. Data can also be reloaded from EEPROM to the scratchpad at any time using the Recall E² [B8h] command. The master can issue “read time slots” (see the 1-WIRE BUS SYSTEM section) following the Recall E² command and the DS18S20 will indicate the status of the recall by transmitting 0 while the recall is in progress and 1 when the recall is done.

DS18S20 MEMORY MAP



CRC GENERATION

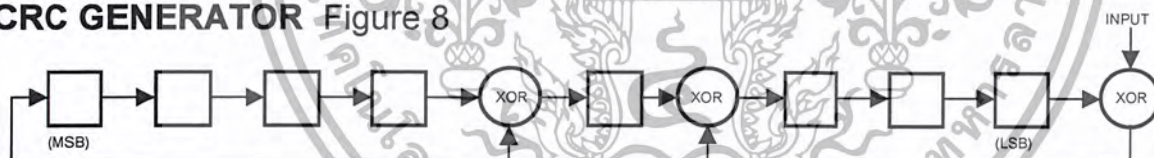
CRC bytes are provided as part of the DS18S20's 64-bit ROM code and in the 9th byte of the scratchpad memory. The ROM code CRC is calculated from the first 56 bits of the ROM code and is contained in the most significant byte of the ROM. The scratchpad CRC is calculated from the data stored in the scratchpad, and therefore it changes when the data in the scratchpad changes. The CRCs provide the bus master with a method of data validation when data is read from the DS18S20. To verify that data has been read correctly, the bus master must re-calculate the CRC from the received data and then compare this value to either the ROM code CRC (for ROM reads) or to the scratchpad CRC (for scratchpad reads). If the calculated CRC matches the read CRC, the data has been received error free. The comparison of CRC values and the decision to continue with an operation are determined entirely by the bus master. There is no circuitry inside the DS18S20 that prevents a command sequence from proceeding if the DS18S20 CRC (ROM or scratchpad) does not match the value generated by the bus master.

The equivalent polynomial function of the CRC (ROM or scratchpad) is:

$$\text{CRC} = X^8 + X^5 + X^4 + 1$$

The bus master can re-calculate the CRC and compare it to the CRC values from the DS18S20 using the polynomial generator shown in Figure 8. This circuit consists of a shift register and XOR gates, and the shift register bits are initialized to 0. Starting with the least significant bit of the ROM code or the least significant bit of byte 0 in the scratchpad, one bit at a time should be shifted into the shift register. After shifting in the 56th bit from the ROM or the most significant bit of byte 7 from the scratchpad, the polynomial generator will contain the re-calculated CRC. Next, the 8-bit ROM code or scratchpad CRC from the DS18S20 must be shifted into the circuit. At this point, if the re-calculated CRC was correct, the shift register will contain all 0s. 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."

CRC GENERATOR Figure 8



1-WIRE BUS SYSTEM

The 1-wire bus system uses a single bus master to control one or more slave devices. The DS18S20 is always a slave. When there is only one slave on the bus, the system is referred to as a “single-drop” system; the system is “multi-drop” if there are multiple slaves on the bus.

All data and commands are transmitted least significant bit first over the 1-wire bus.

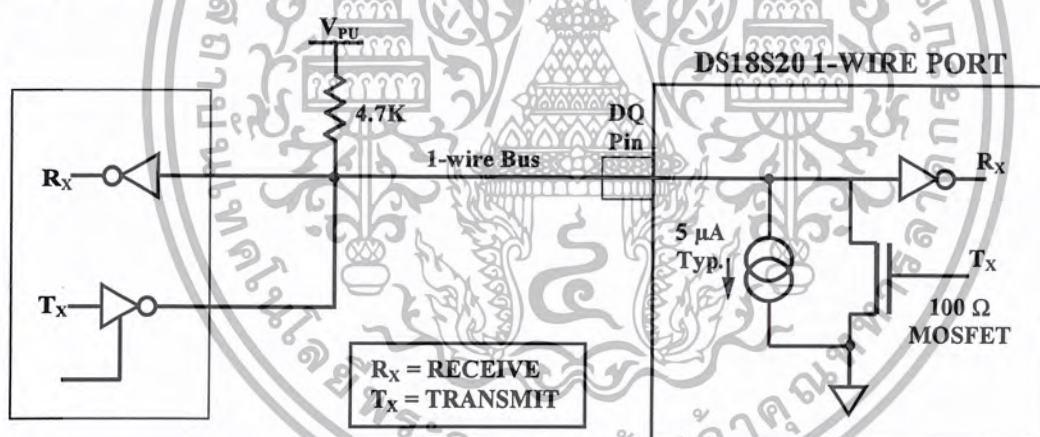
The following discussion of the 1-wire bus system is broken down into three topics: hardware configuration, transaction sequence, and 1-wire signaling (signal types and timing).

HARDWARE CONFIGURATION

The 1-wire bus has by definition only a single data line. Each device (master or slave) interfaces to the data line via an open drain or 3-state port. This allows each device to “release” the data line when the device is not transmitting data so the bus is available for use by another device. The 1-wire port of the DS18S20 (the DQ pin) is open drain with an internal circuit equivalent to that shown in Figure 9.

The 1-wire bus requires an external pullup resistor of approximately 5 k Ω ; thus, 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 the bus is held low for more than 480 μ s, all components on the bus will be reset.

HARDWARE CONFIGURATION Figure 9



TRANSACTION SEQUENCE

The transaction sequence for accessing the DS18S20 is as follows:

Step 1. Initialization

Step 2. ROM Command (followed by any required data exchange)

Step 3. DS18S20 Function Command (followed by any required data exchange)

It is very important to follow this sequence every time the DS18S20 is accessed, as the DS18S20 will not respond if any steps in the sequence are missing or out of order. Exceptions to this rule are the Search ROM [F0h] and Alarm Search [ECh] commands. After issuing either of these ROM commands, the master must return to Step 1 in the sequence.

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 slave devices (such as the DS18S20) are on the bus and are ready to operate. Timing for the reset and presence pulses is detailed in the 1-WIRE SIGNALING section.

ROM COMMANDS

After the bus master has detected a presence pulse, it can issue a ROM command. These commands operate on the unique 64-bit ROM codes of each slave device and allow the master to single out a specific device if many are present on the 1-wire bus. These commands also allow the master to determine how many and what types of devices are present on the bus or if any device has experienced an alarm condition. There are five ROM commands, and each command is 8 bits long. The master device must issue an appropriate ROM command before issuing a DS18S20 function command. A flowchart for operation of the ROM commands is shown in Figure 14.

SEARCH ROM [F0h]

When a system is initially powered up, the master must identify the ROM codes of all slave devices on the bus, which allows the master to determine the number of slaves and their device types. The master learns the ROM codes through a process of elimination that requires the master to perform a Search ROM cycle (i.e., Search ROM command followed by data exchange) as many times as necessary to identify all of the slave devices. If there is only one slave on the bus, the simpler Read ROM command (see below) can be used in place of the Search ROM process. For a detailed explanation of the Search ROM procedure, refer to the iButton Book of Standards at www.ibutton.com/ibuttons/standard.pdf. After every Search ROM cycle, the bus master must return to Step 1 (Initialization) in the transaction sequence.

READ ROM [33h]

This command can only be used when there is one slave on the bus. It allows the bus master to read the slave's 64-bit ROM code without using the Search ROM procedure. If this command is used when there is more than one slave present on the bus, a data collision will occur when all the slaves attempt to respond at the same time.

MATCH ROM [55h]

The match ROM command followed by a 64-bit ROM code sequence allows the bus master to address a specific slave device on a multi-drop or single-drop bus. Only the slave that exactly matches the 64-bit ROM code sequence will respond to the function command issued by the master; all other slaves on the bus will wait for a reset pulse.

SKIP ROM [CCh]

The master can use this command to address all devices on the bus simultaneously without sending out any ROM code information. For example, the master can make all DS18S20s on the bus perform simultaneous temperature conversions by issuing a Skip ROM command followed by a Convert T [44h] command. Note, however, that the Skip ROM command can only be followed by the Read Scratchpad [BEh] command when there is one slave on the bus. This sequence saves time by allowing the master to read from the device without sending its 64-bit ROM code. This sequence will cause a data collision on the bus if there is more than one slave since multiple devices will attempt to transmit data simultaneously.

ALARM SEARCH [ECh]

The operation of this command is identical to the operation of the Search ROM command except that only slaves with a set alarm flag will respond. This command allows the master device to determine if any DS18S20s experienced an alarm condition during the most recent temperature conversion. After

every Alarm Search cycle (i.e., Alarm Search command followed by data exchange), the bus master must return to Step 1 (Initialization) in the transaction sequence. Refer to the OPERATION – ALARM SIGNALING section for an explanation of alarm flag operation.

DS18S20 FUNCTION COMMANDS

After the bus master has used a ROM command to address the DS18S20 with which it wishes to communicate, the master can issue one of the DS18S20 function commands. These commands allow the master to write to and read from the DS18S20's scratchpad memory, initiate temperature conversions and determine the power supply mode. The DS18S20 function commands, which are described below, are summarized in Table 4 and illustrated by the flowchart in Figure 15.

CONVERT T [44h]

This command initiates a single temperature conversion. Following the conversion, the resulting thermal data is stored in the 2-byte temperature register in the scratchpad memory and the DS18S20 returns to its low-power idle state. If the device is being used in parasite power mode, within 10 μ s (max) after this command is issued the master must enable a strong pullup on the 1-wire bus for the duration of the conversion (t_{conv}) as described in the POWERING THE DS18S20 section. If the DS18S20 is powered by an external supply, the master can issue read time slots after the Convert T command and the DS18S20 will respond by transmitting 0 while the temperature conversion is in progress and 1 when the conversion is done. In parasite power mode this notification technique cannot be used since the bus is pulled high by the strong pullup during the conversion.

WRITE SCRATCHPAD [4Eh]

This command allows the master to write 2 bytes of data to the DS18S20's scratchpad. The first byte is written into the T_H register (byte 2 of the scratchpad), and the second byte is written into the T_L register (byte 3 of the scratchpad). Data must be transmitted least significant bit first. Both bytes MUST be written before the master issues a reset, or the data may be corrupted.

READ SCRATCHPAD [BEh]

This command allows the master to read the contents of the scratchpad. The data transfer starts with the least significant bit of byte 0 and continues through the scratchpad until the 9th byte (byte 8 – CRC) is read. The master may issue a reset to terminate reading at any time if only part of the scratchpad data is needed.

COPY SCRATCHPAD [48h]

This command copies the contents of the scratchpad T_H and T_L registers (bytes 2 and 3) to EEPROM. If the device is being used in parasite power mode, within 10 μ s (max) after this command is issued the master must enable a strong pullup on the 1-wire bus for at least 10 ms as described in the POWERING THE DS18S20 section.

RECALL E² [B8h]

This command recalls the alarm trigger values (T_H and T_L) from EEPROM and places the data in bytes 2 and 3, respectively, in the scratchpad memory. The master device can issue read time slots following the Recall E² command and the DS18S20 will indicate the status of the recall by transmitting 0 while the recall is in progress and 1 when the recall is done. The recall operation happens automatically at power-up, so valid data is available in the scratchpad as soon as power is applied to the device.

READ POWER SUPPLY [B4h]

The master device issues this command followed by a read time slot to determine if any DS18S20s on the bus are using parasite power. During the read time slot, parasite powered DS18S20s will pull the bus low, and externally powered DS18S20s will let the bus remain high. Refer to the POWERING THE DS18S20 section for usage information for this command.

DS18S20 FUNCTION COMMAND SET Table 4

Command	Description	Protocol	1-Wire Bus Activity After Command is Issued	Notes
TEMPERATURE CONVERSION COMMANDS				
Convert T	Initiates temperature conversion.	44h	DS18S20 transmits conversion status to master (not applicable for parasite-powered DS18S20s).	1
MEMORY COMMANDS				
Read Scratchpad	Reads the entire scratchpad including the CRC byte.	BEh	DS18S20 transmits up to 9 data bytes to master.	2
Write Scratchpad	Writes data into scratchpad bytes 2 and 3 (T_H and T_L).	4Eh	Master transmits 2 data bytes to DS18S20.	3
Copy Scratchpad	Copies T_H and T_L data from the scratchpad to EEPROM.	48h	None	1
Recall E^2	Recalls T_H and T_L data from EEPROM to the scratchpad.	B8h	DS18S20 transmits recall status to master.	
Read Power Supply	Signals DS18S20 power supply mode to the master.	B4h	DS18S20 transmits supply status to master.	

NOTES:

1. For parasite-powered DS18S20s, the master must enable a strong pullup on the 1-wire bus during temperature conversions and copies from the scratchpad to EEPROM. No other bus activity may take place during this time.
2. The master can interrupt the transmission of data at any time by issuing a reset.
3. Both bytes must be written before a reset is issued.

1-WIRE SIGNALING

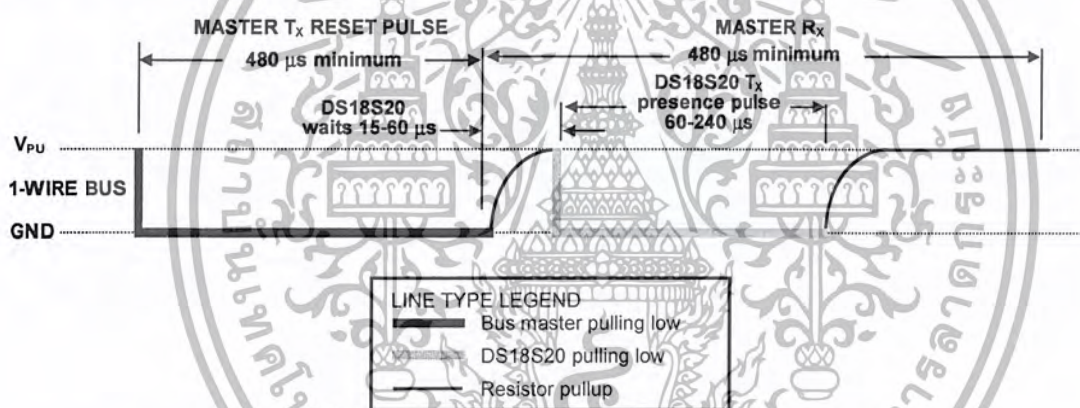
The DS18S20 uses a strict 1-wire communication protocol to insure data integrity. Several signal types are defined by this protocol: reset pulse, presence pulse, write 0, write 1, read 0, and read 1. All of these signals, with the exception of the presence pulse, are initiated by the bus master.

INITIALIZATION PROCEDURE: RESET AND PRESENCE PULSES

All communication with the DS18S20 begins with an initialization sequence that consists of a reset pulse from the master followed by a presence pulse from the DS18S20. This is illustrated in Figure 10. When the DS18S20 sends the presence pulse in response to the reset, it is indicating to the master that it is on the bus and ready to operate.

During the initialization sequence the bus master transmits (T_X) the reset pulse by pulling the 1-wire bus low for a minimum of 480 μs . The bus master then releases the bus and goes into receive mode (R_X). When the bus is released, the 5k pullup resistor pulls the 1-wire bus high. When the DS18S20 detects this rising edge, it waits 15–60 μs and then transmits a presence pulse by pulling the 1-wire bus low for 60–240 μs .

INITIALIZATION TIMING Figure 10



READ/WRITE TIME SLOTS

The bus master writes data to the DS18S20 during write time slots and reads data from the DS18S20 during read time slots. One bit of data is transmitted over the 1-wire bus per time slot.

WRITE TIME SLOTS

There are two types of write time slots: “Write 1” time slots and “Write 0” time slots. The bus master uses a Write 1 time slot to write a logic 1 to the DS18S20 and a Write 0 time slot to write a logic 0 to the DS18S20. All write time slots must be a minimum of 60 μs in duration with a minimum of a 1 μs recovery time between individual write slots. Both types of write time slots are initiated by the master pulling the 1-wire bus low (see Figure 11).

To generate a Write 1 time slot, after pulling the 1-wire bus low, the bus master must release the 1-wire bus within 15 μs . When the bus is released, the 5k pullup resistor will pull the bus high. To generate a Write 0 time slot, after pulling the 1-wire bus low, the bus master must continue to hold the bus low for the duration of the time slot (at least 60 μs). The DS18S20 samples the 1-wire bus during a window that lasts from 15 μs to 60 μs after the master initiates the write time slot. If the bus is high during the sampling window, a 1 is written to the DS18S20. If the line is low, a 0 is written to the DS18S20.

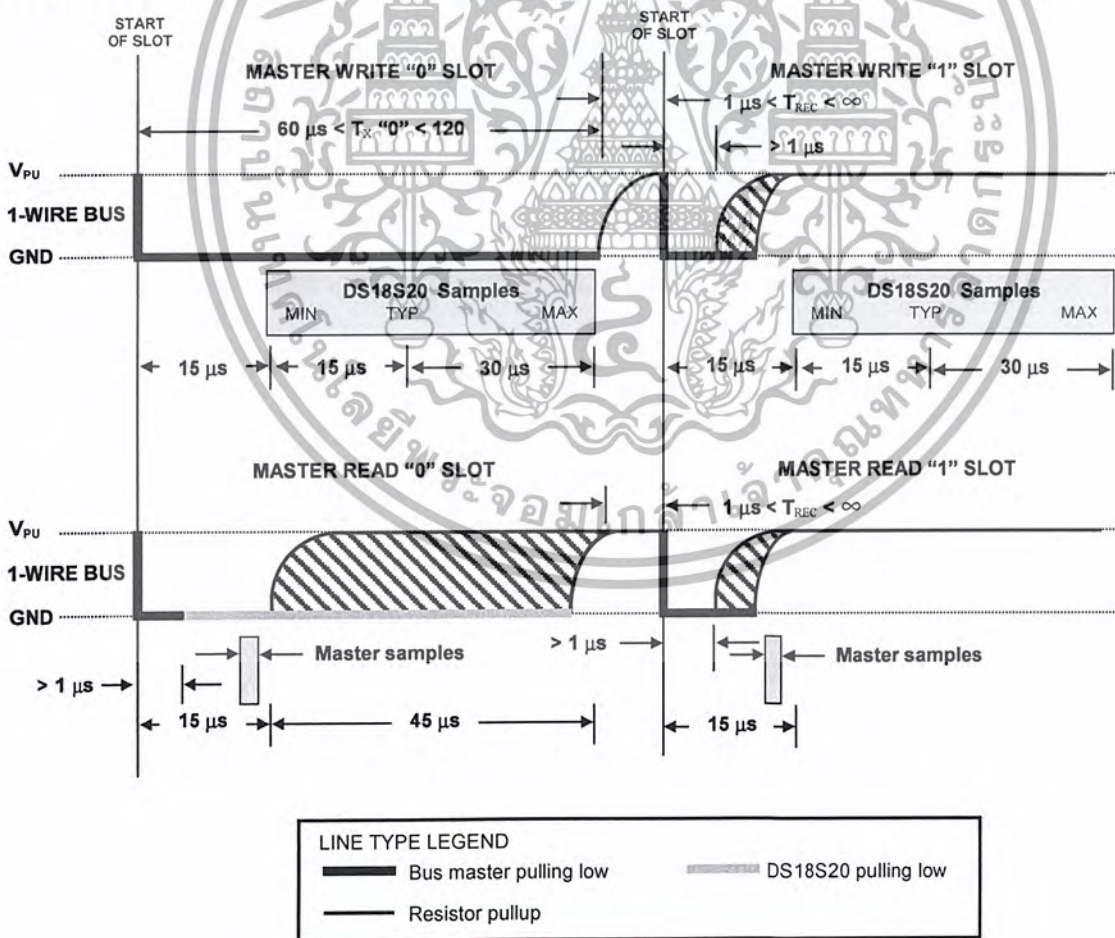
READ TIME SLOTS

The DS18S20 can only transmit data to the master when the master issues read time slots. Therefore, the master must generate read time slots immediately after issuing a Read Scratchpad [BEh] or Read Power Supply [B4h] command, so that the DS18S20 can provide the requested data. In addition, the master can generate read time slots after issuing Convert T [44h] or Recall E² [B8h] commands to find out the status of the operation as explained in the DS18S20 FUNCTION COMMAND section.

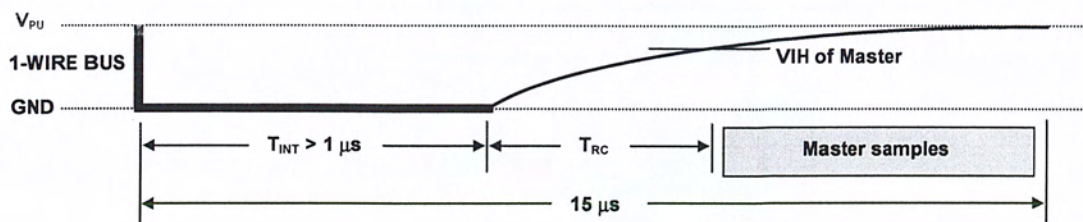
All read time slots must be a minimum of 60 μs in duration with a minimum of a 1 μs recovery time between slots. A read time slot is initiated by the master device pulling the 1-wire bus low for a minimum of 1 μs and then releasing the bus (see Figure 11). After the master initiates the read time slot, the DS18S20 will begin transmitting a 1 or 0 on bus. The DS18S20 transmits a 1 by leaving the bus high and transmits a 0 by pulling the bus low. When transmitting a 0, the DS18S20 will release the bus by the end of the time slot, and the bus will be pulled back to its high idle state by the pullup resistor. Output data from the DS18S20 is valid for 15 μs after the falling edge that initiated the read time slot. Therefore, the master must release the bus and then sample the bus state within 15 μs from the start of the slot.

Figure 12 illustrates that the sum of T_{INIT}, T_{RC}, and T_{SAMPLE} must be less than 15 μs for a read time slot. Figure 13 shows that system timing margin is maximized by keeping T_{INIT} and T_{RC} as short as possible and by locating the master sample time during read time slots towards the end of the 15 μs period.

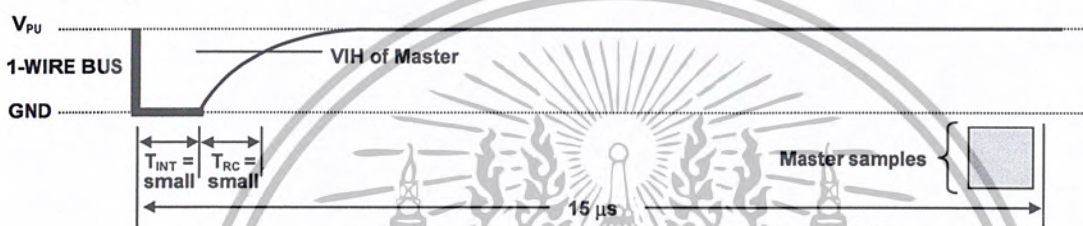
READ/WRITE TIME SLOT TIMING DIAGRAM Figure 11





DETAILED MASTER READ 1 TIMING Figure 12



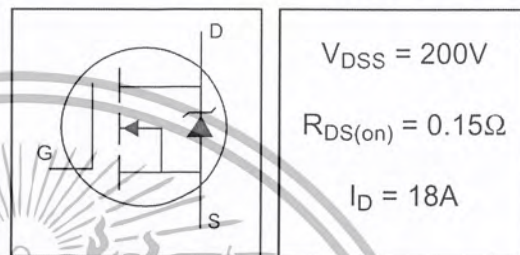
RECOMMENDED MASTER READ 1 TIMING Figure 13



LINE TYPE LEGEND
 Bus master pulling low
 Resistor pullup

- Advanced Process Technology
- Dynamic dv/dt Rating
- 175°C Operating Temperature
- Fast Switching
- Fully Avalanche Rated
- Ease of Paralleling
- Simple Drive Requirements

HEXFET® Power MOSFET



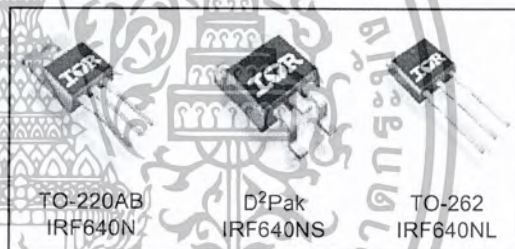
Description

Fifth Generation HEXFET® Power MOSFETs from International Rectifier utilize advanced processing techniques to achieve extremely low on-resistance per silicon area. This benefit, combined with the fast switching speed and ruggedized device design that HEXFET Power MOSFETs are well known for, provides the designer with an extremely efficient and reliable device for use in a wide variety of applications.

The TO-220 package is universally preferred for all commercial-industrial applications at power dissipation levels to approximately 50 watts. The low thermal resistance and low package cost of the TO-220 contribute to its wide acceptance throughout the industry.

The D²Pak is a surface mount power package capable of accommodating die sizes up to HEX-4. It provides the highest power capability and the lowest possible on-resistance in any existing surface mount package. The D²Pak is suitable for high current applications because of its low internal connection resistance and can dissipate up to 2.0W in a typical surface mount application.

The through-hole version (IRF640NL) is available for low-profile application.



Absolute Maximum Ratings

	Parameter	Max.	Units
$I_D @ T_C = 25^\circ C$	Continuous Drain Current, $V_{GS} @ 10V$	18	A
$I_D @ T_C = 100^\circ C$	Continuous Drain Current, $V_{GS} @ 10V$	13	
I_{DM}	Pulsed Drain Current ①	72	
$P_D @ T_C = 25^\circ C$	Power Dissipation	150	W
	Linear Derating Factor	1.0	W/°C
V_{GS}	Gate-to-Source Voltage	± 20	V
E_{AS}	Single Pulse Avalanche Energy ②	247	mJ
I_{AR}	Avalanche Current ③	18	A
E_{AR}	Repetitive Avalanche Energy ③	15	mJ
dv/dt	Peak Diode Recovery dv/dt ④	8.1	V/ns
T_J	Operating Junction and	-55 to +175	°C
T_{STG}	Storage Temperature Range		
	Soldering Temperature, for 10 seconds	300 (1.6mm from case)	
	Mounting torque, 6-32 or M3 screw ⑤	10 lb•in (1.1N•m)	

IRF640N/S/L

International
IR Rectifier

Electrical Characteristics @ T_J = 25°C (unless otherwise specified)

	Parameter	Min.	Typ.	Max.	Units	Conditions
V _{(BR)DSS}	Drain-to-Source Breakdown Voltage	200	—	—	V	V _{GS} = 0V, I _D = 250μA
ΔV _{(BR)DSS/ΔT_J}	Breakdown Voltage Temp. Coefficient	—	0.25	—	V/°C	Reference to 25°C, I _D = 1mA
R _{DS(on)}	Static Drain-to-Source On-Resistance	—	—	0.15	Ω	V _{GS} = 10V, I _D = 11A ③
V _{GS(th)}	Gate Threshold Voltage	2.0	—	4.0	V	V _{DS} = V _{GS} , I _D = 250μA
g _{fs}	Forward Transconductance	6.8	—	—	S	V _{DS} = 50V, I _D = 11A ③
I _{DSS}	Drain-to-Source Leakage Current	—	—	25	μA	V _{DS} = 200V, V _{GS} = 0V
		—	—	250	μA	V _{DS} = 160V, V _{GS} = 0V, T _J = 150°C
I _{GSS}	Gate-to-Source Forward Leakage	—	—	100	nA	V _{GS} = 20V
	Gate-to-Source Reverse Leakage	—	—	-100	nA	V _{GS} = -20V
Q _g	Total Gate Charge	—	—	67	nC	I _D = 11A
Q _{gs}	Gate-to-Source Charge	—	—	11	nC	V _{DS} = 160V
Q _{gd}	Gate-to-Drain ("Miller") Charge	—	—	33	nC	V _{GS} = 10V, See Fig. 6 and 13
t _{d(on)}	Turn-On Delay Time	—	10	—	ns	V _{DD} = 100V
t _r	Rise Time	—	19	—	ns	I _D = 11A
t _{d(off)}	Turn-Off Delay Time	—	23	—	ns	R _G = 2.5Ω
t _f	Fall Time	—	5.5	—	ns	R _D = 9.0Ω, See Fig. 10 ③
L _D	Internal Drain Inductance	—	4.5	—	nH	Between lead, 6mm (0.25in.) from package and center of die contact
L _S	Internal Source Inductance	—	7.5	—	nH	
C _{iss}	Input Capacitance	—	1160	—	pF	V _{GS} = 0V
C _{oss}	Output Capacitance	—	185	—	pF	V _{DS} = 25V
C _{rss}	Reverse Transfer Capacitance	—	53	—	pF	f = 1.0MHz, See Fig. 5

Source-Drain Ratings and Characteristics

	Parameter	Min.	Typ.	Max.	Units	Conditions
I _S	Continuous Source Current (Body Diode)	—	—	18	A	MOSFET symbol showing the integral reverse p-n junction diode.
I _{SM}	Pulsed Source Current (Body Diode)①	—	—	72	A	
V _{SD}	Diode Forward Voltage	—	—	1.3	V	T _J = 25°C, I _S = 11A, V _{GS} = 0V ③
t _{rr}	Reverse Recovery Time	—	167	251	ns	T _J = 25°C, I _F = 11A
Q _{rr}	Reverse Recovery Charge	—	929	1394	nC	di/dt = 100A/μs ③
t _{on}	Forward Turn-On Time	Intrinsic turn-on time is negligible (turn-on is dominated by L _S +L _D)				

Thermal Resistance

	Parameter	Typ.	Max.	Units
R _{θJC}	Junction-to-Case	—	1.0	°C/W
R _{θCS}	Case-to-Sink, Flat, Greased Surface ④	0.50	—	
R _{θJA}	Junction-to-Ambient④	—	62	
R _{θJA}	Junction-to-Ambient (PCB mount)⑤	—	40	

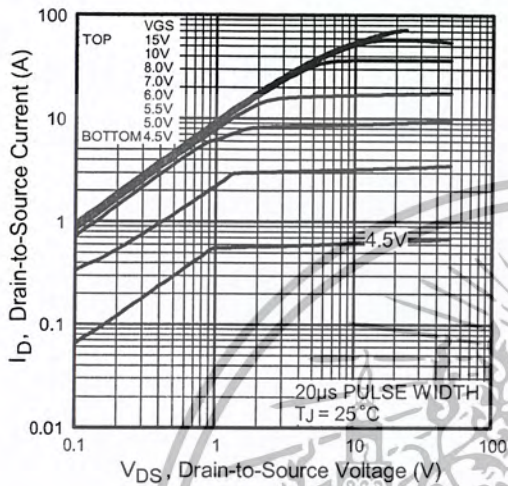


Fig 1. Typical Output Characteristics

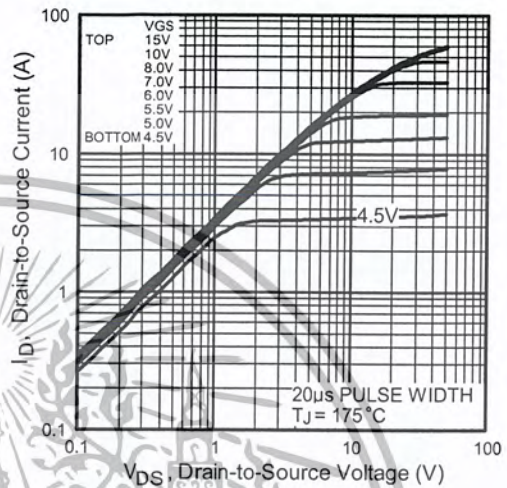


Fig 2. Typical Output Characteristics

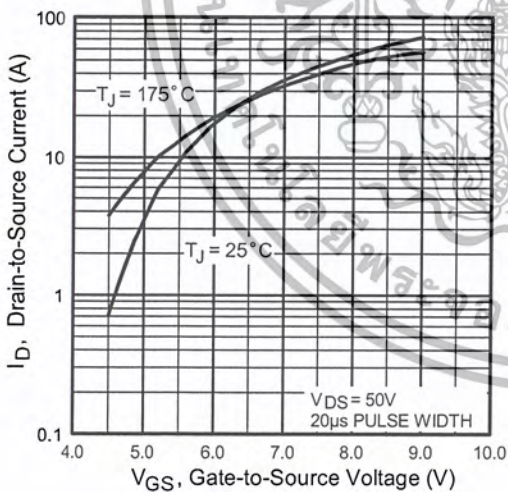


Fig 3. Typical Transfer Characteristics

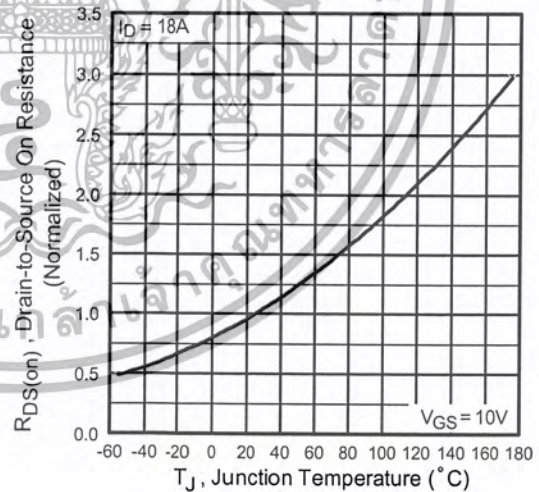


Fig 4. Normalized On-Resistance Vs. Temperature

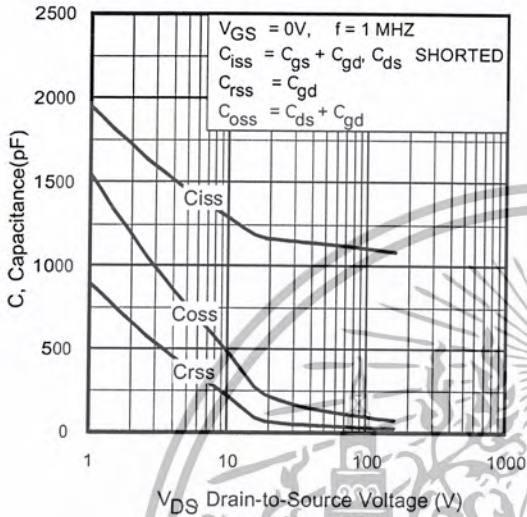


Fig 5. Typical Capacitance Vs. Drain-to-Source Voltage

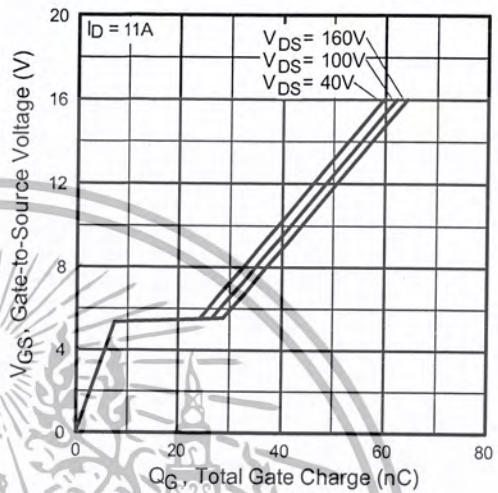


Fig 6. Typical Gate Charge Vs. Gate-to-Source Voltage

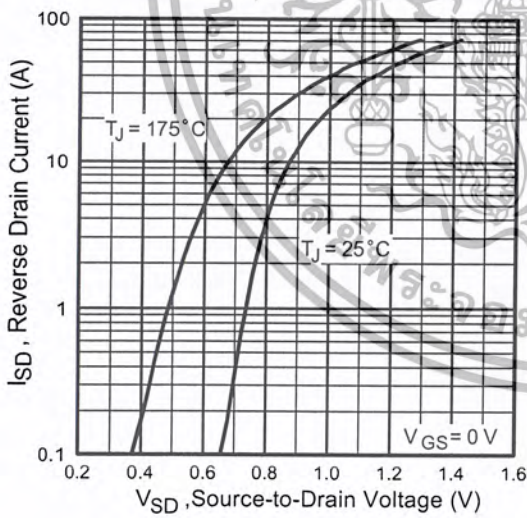


Fig 7. Typical Source-Drain Diode Forward Voltage

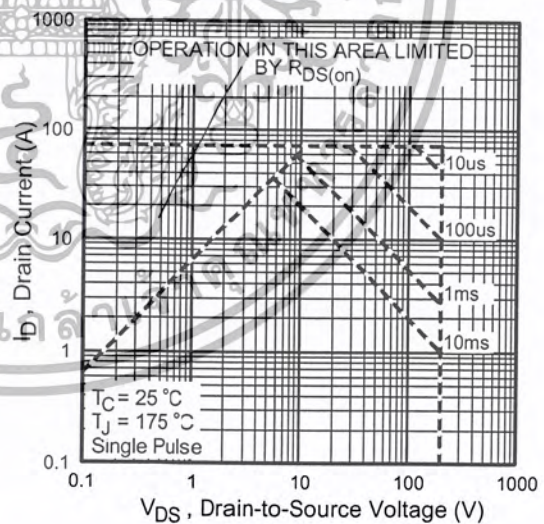


Fig 8. Maximum Safe Operating Area

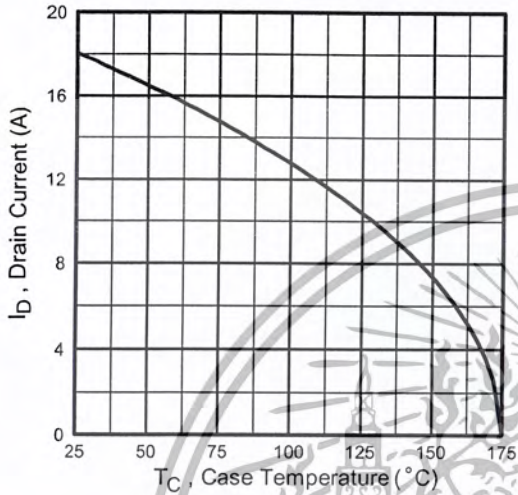


Fig 9. Maximum Drain Current Vs. Case Temperature

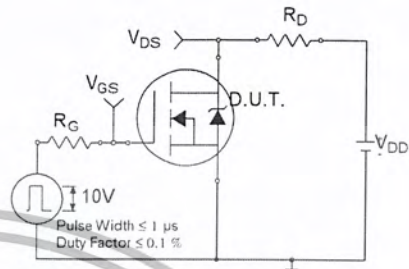


Fig 10a. Switching Time Test Circuit

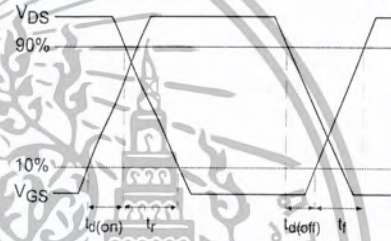


Fig 10b. Switching Time Waveforms

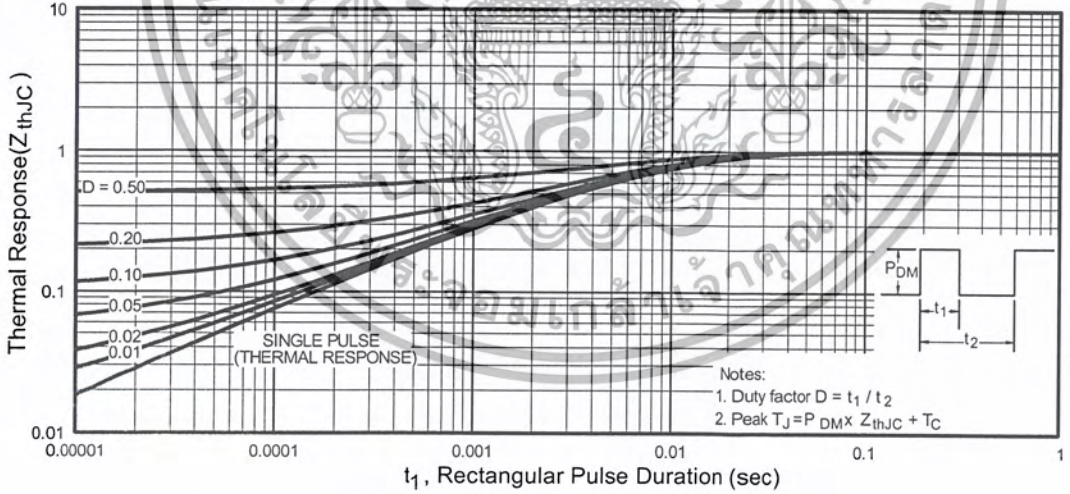


Fig 11. Maximum Effective Transient Thermal Impedance, Junction-to-Case

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IRF640N/S/L

International
IR Rectifier

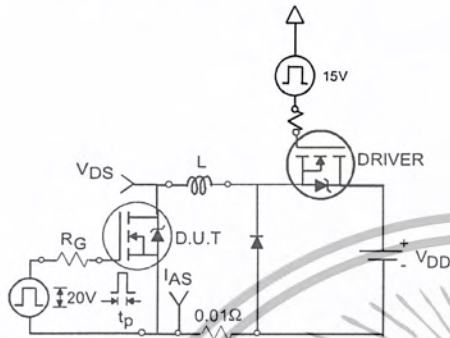


Fig 12a. Unclamped Inductive Test Circuit

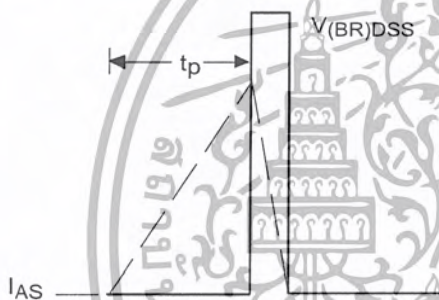


Fig 12b. Unclamped Inductive Waveforms

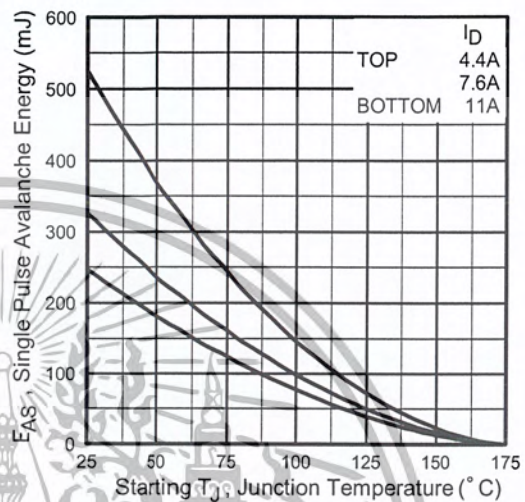


Fig 12c. Maximum Avalanche Energy Vs. Drain Current

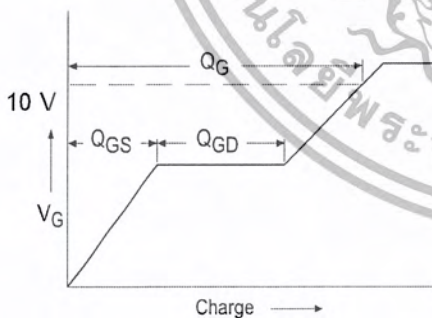


Fig 13a. Basic Gate Charge Waveform

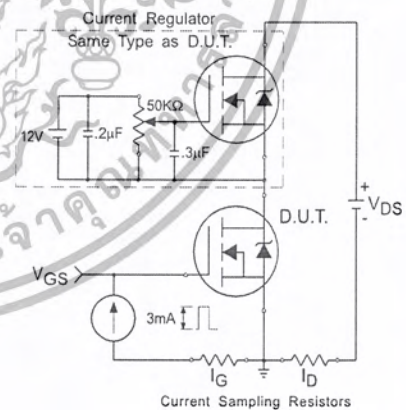


Fig 13b. Gate Charge Test Circuit

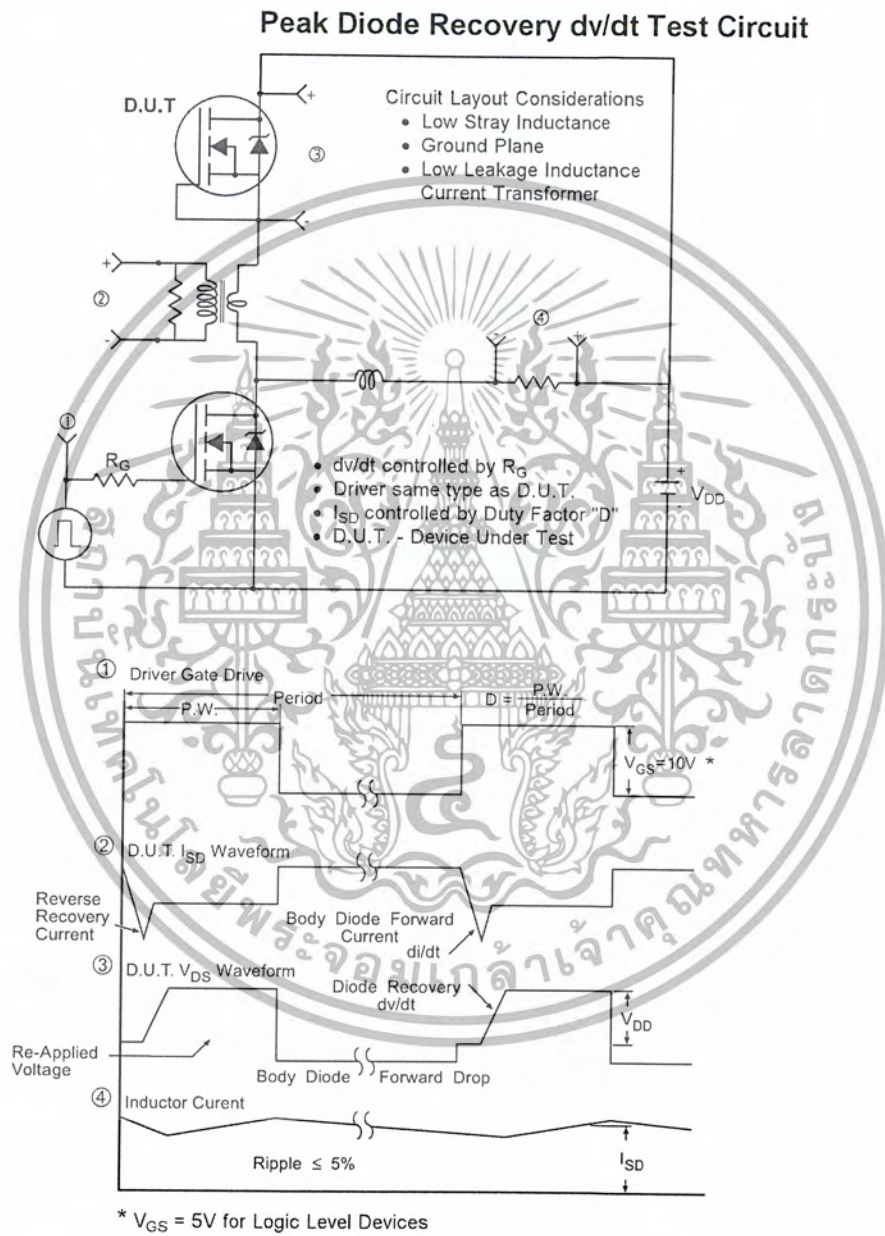


Fig 14. For N-Channel HEXFET® Power MOSFETs

```
#include <16F877A.h>
```

```
#fuses HS,NOWDT,NOPROTECT,NOLVP
```

```
#use delay(clock=20000000)
```

```
#include <TOUCH.c>
```

```
#include <lcd.c>
```

```
#include <math.h>
```

```
#int_CCP1
```

```
float deg;
```

```
float deg1;
```

```
float deg2;
```

```
float deg3;
```

```
/****** ReadTemp_DS1820 */
```

```
void ReadTemp_DS1820(void)
```

```
{
```

```
byte i, buffer[9];
```

```
int count=1;
```

```
if (touch_present())
```

```
{
```

```
touch_write_byte(0xCC);
```

```
touch_write_byte (0x44);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay_ms(200);

touch_present();

touch_write_byte(0xCC);

touch_write_byte (0xBE);

for(i=0; i<9;i++)

    buffer[i] = touch_read_byte();

}

if(count==1){

deg=(float)buffer[0]/2;

if(deg<80)

{

printf(lcd_putc, "\ftemp = %3.2f c", deg);

printf(lcd_putc, "\n LEFT PART");

}

//delay_ms(400);

delay_ms(1500);

}

}

void Readtemp2_DS1820(void)

```



```

{

byte j,buffer[9];

int count=1;

if(touch_present1()){

touch_write_byte1(0xCC);

touch_write_byte1(0x44);

delay_ms(750);

touch_present1();

touch_write_byte1(0xCC);

touch_write_byte1(0xBE);

for(j=0;j<9;j++)

    buffer[j] = touch_read_byte1();

}

if(count==1){

deg1=(float)buffer[0]/2;

printf(lcd_putc,"\ftemp = %3.2f c",deg1);

printf(lcd_putc,"\n FRONT PART");

delay_ms(1500);

```



```

}

}

void Readtemp3_DS1820(void)

{

byte k,buffer[9];

int count=1;

if(touch_present2()){

touch_write_byte2(0xCC);

touch_write_byte2(0x44);

delay_ms(750);

touch_present2();

touch_write_byte2(0xCC);

touch_write_byte2(0xBE);

for(k=0;k<9;k++)

buffer[k] = touch_read_byte2();

}

if(count==1){

deg2=(float)buffer[0]/2;

printf(lcd_putc,"\ftemp = %3.2f c",deg2);

```



```
printf(lcd_putc, "\n BACK PART");
```

```
delay_ms(1500);
```

```
}
```

```
}
```

```
void Readtemp4_DS1820(void)
```

```
{
```

```
byte l,buffer[9];
```

```
int count=1;
```

```
if(touch_present3()){
```

```
touch_write_byte3(0xCC);
```

```
touch_write_byte3(0x44);
```

```
delay_ms(750);
```

```
touch_present3();
```

```
touch_write_byte3(0xCC);
```

```
touch_write_byte3(0xBE);
```

```
for(l=0;l<9;l++)
```

```
buffer[l] = touch_read_byte3();
```

```
}
```

```
if(count==1){
```

```
deg3=(float)buffer[0]/2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

printf(lcd_putc, "\ftemp = %3.2f c", deg3);

printf(lcd_putc, "\n RIGHT PART");

delay_ms(1500);

}

}

```

```

/***** Main Functions */

```

```

void main(void)

```

```

{

```

```

char count=0;

```

```

float a=37.00,c,d,dm;

```

```

int b=0,z=0;

```

```

unsigned int16 duty=0 ;

```

```

set_tris_C(0x00);

```

```

set_tris_A(0x0F);

```

```

enable_interrupts(GLOBAL);

```

```

enable_interrupts(INT_CCP1);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setup_CCPI(CCP_PWM);
```

```
setup_timer_2(T2_DIV_BY_16,255,1);
```

```
set_timer2(0);
```

```
set_pwm1_duty(duty);
```

```
lcd_init();
```

```
port_b_pullups(true);
```

```
lcd_gotoxy(5,1);
```

```
lcd_putc("WELCOME");
```

```
delay_ms(1000);
```

```
lcd_putc("\nTemp Simulation");
```

```
delay_ms(1000);
```

```
lcd_gotoxy(3,1);
```

```
lcd_putc("\fHYPOTHERMIA");
```

```
delay_ms(1000);
```

```
lcd_gotoxy(5,1);
```

```
lcd_putc("\nHELMET");
```

```
delay_ms(1000);
```

```
lcd_gotoxy(5,1);
```

```
lcd_putc("\fAdvisor");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
delay_ms(800);
```

```
lcd_gotoxy(2,2);
```

```
lcd_putc("\nDr.KITTIPOL");
```

```
delay_ms(1500);
```

```
lcd_putc("\fDesign By");
```

```
delay_ms(1000);
```

```
lcd_putc("\fJATUPORN 0189");
```

```
delay_ms(800);
```

```
lcd_putc("\nJAKHAPONG 0203");
```

```
delay_ms(1000);
```

```
lcd_putc("\fCHUTSIRIN 0276");
```

```
delay_ms(800);
```

```
lcd_putc("\nCHALERMPOLO 0283");
```

```
delay_ms(1000);
```

```
while(TRUE){
```

```
ReadTemp_DS1820();
```

```
if(deg<80)
```

```
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ReadTemp2_DS1820();
```

```
ReadTemp3_DS1820();
```

```
ReadTemp4_DS1820();
```

```
}
```

```
if(deg>deg1)
```

```
{
```

```
dm=deg;
```

```
}
```

```
else
```

```
{
```

```
dm=deg1;
```

```
}
```

```
if(deg2>dm)
```

```
{
```

```
dm=deg2;
```

```
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(deg3>dm)
```

```
{
```

```
dm=deg3;
```

```
}
```

```
if(dm<80)
```

```
{
```

```
if((dm<37.5)&&(dm>=37))
```

```
//(((deg<37.5)&&(deg>=37))||((deg1<37.5)&&(deg1>=37))||((deg2<37.5)&&(deg2>=37))||((deg3<37.5)  
&&(deg3>=37)))
```

```
{
```

```
//output_high(PIN_C4); // output pin_C4 is high
```

```
duty =256; //64
```

```
set_pwm1_duty(duty); // PIN_C2 = 25%
```

```
lcd_gotoxy(1,1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{lcd_putc("\fLevel 1\nDuty Cycle 25%\n");}

delay_ms(1500);

}

else if((dm<38.5)&&(dm>=37.5))
//(((deg<38.5)&&(deg>=37.5))||((deg1<38.5)&&(deg1>=37.5))||((deg2<38.5)&&(deg2>=37.5))||((deg3<3
8.5)&&(deg3>=37.5)))
{
//output_high(PIN_C4); // output pin_C4 is high

duty =512; // 128
set_pwm1_duty(duty); // PIN_C2 = 50%

lcd_gotoxy(1,1);
{lcd_putc("\fLevel 2\nDuty Cycle 50%\n");}

delay_ms(1500);

}

else if((dm<39)&&(dm>=38.5))
//(((deg<39)&&(deg>=38.5))||((deg1<39)&&(deg1>=38.5))||((deg2<39)&&(deg2>=38.5))||((deg3<39)&
&(deg3>=38.5)))
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//output_high(PIN_C4);      // output pin_C4 is high

duty =768; //192

set_pwm1_duty(duty);// PIN_C2 = 75%

lcd_gotoxy(1,1);

{lcd_putc("\fLevel 3\nDuty Cycle 75%\n");}

delay_ms(1500);
}

else if (dm>=39) //((deg>=39)||((deg1>=39)||((deg2>=39)||((deg3>=39)))

{

//output_high(PIN_C4);      // output pin_C4 is high

duty =1023;

set_pwm1_duty(duty);// PIN_C2 = 100%

lcd_gotoxy(1,1);

{lcd_putc("\fLevel 4\nDuty Cycle 100%\n");}

delay_ms(1500);

}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else{  
  
//output_high(PIN_C4);      // output pin_C4 is high  
  
duty =0;  
  
set_pwm1_duty(duty); // PIN_C2 = 0% >>>> Motor Stop !!
```

```
lcd_gotoxy(1,1);
```

```
{lcd_putc("\fMotor stop!!\n");}
```

```
delay_ms(1500);
```

