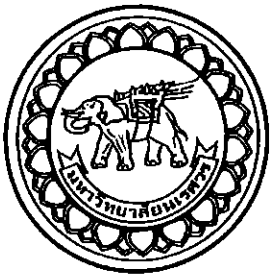


# การควบคุมอุณหภูมิของตู้เลี้ยงปลาโดยใช้ไมโครคอนโทรลเลอร์

# MICROCONTROLLER-BASED AQDUARIUM TEMPERATURE CONTROL


นายณัฐพล ชอบจิตต์ รหัส 50364546

นายธีรพงศ์ มีแม่นวิทย์ รหัส 50364607



ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 19 มี.ค. 2555
เลขทะเบียน 15747164
เลขเรียกหนังสือ ปง.
มหาวิทยาลัยนเรศวร ล4342 ป
2553


ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2553

# ใบรับรองปริญญานิพนธ์

**ชื่อหัวข้อโครงงาน** การควบคุมอุณหภูมิของตู้เลี้ยงปลาโดยใช้ไมโครคอนโทรลเลอร์

**ผู้ดำเนินโครงงาน** นายณัฐพล ชอบจิตต์ รหัส 50364546

นายธีรพงศ์ มีแม่นวิทย์ รหัส 50364607

**ที่ปรึกษาโครงงาน** ดร. นิพัทธ์ จันทรมินทร์

**สาขาวิชา** วิศวกรรมไฟฟ้า

**ภาควิชา** วิศวกรรมไฟฟ้าและคอมพิวเตอร์

**ปีการศึกษา** 2553

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

.....ที่ปรึกษาโครงงาน

(ดร. นิพัทธ์ จันทรมินทร์)

.....กรรมการ

(ดร. ศุภวรรณ พลพิทักษ์ชัย)

.....กรรมการ

(ดร. แคทรียา สุวรรณศรี)

| | |
|---|---|
| **ชื่อหัวข้อโครงงาน** | การควบคุมอุณหภูมิของตู้เลี้ยงปลาโดยใช้ไมโครคอนโทรลเลอร์ |
| **ผู้ดำเนินโครงงาน** | นายณัฐพล ชอบจิตต์ รหัส 50364546 |
| | นายธีรพงศ์ มีแม่นวิทย์ รหัส 50364607 |
| **ที่ปรึกษาโครงงาน** | ดร. นิพัทธ์ จันทรมินทร์ |
| **สาขาวิชา** | วิศวกรรมไฟฟ้า |
| **ภาควิชา** | วิศวกรรมไฟฟ้าและคอมพิวเตอร์ |
| **ปีการศึกษา** | 2553 |


## บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอโครงงานเกี่ยวกับการรักษาอุณหภูมิของน้ำภายในตู้ปลาสวยงาม แนวคิดในการดำเนินโครงงานเนื่องมาจากปัจจุบันการเลี้ยงปลาสวยงามเป็นที่นิยมกันอย่างแพร่หลาย โดยอาจทำเป็นงานอดิเรกหรือทำเป็นอาชีพ แต่เนื่องด้วยสภาพอากาศในประเทศไทยอาจไม่เหมาะสมกับการเลี้ยงปลาสวยงามบางชนิดที่นำเข้าจากต่างประเทศ จึงต้องมีการปรับอุณหภูมิของน้ำให้เหมาะสมกับปลาชนิดนั้นก่อนนำมาเลี้ยงโดยใช้อุปกรณ์ปรับอุณหภูมิ ได้แก่ พัดลมระบายอากาศเพื่อลดอุณหภูมิของน้ำ และเครื่องทำความร้อนเพื่อเพิ่มอุณหภูมิของน้ำ อย่างไรก็ตามอุปกรณ์ดังกล่าวทำหน้าที่เพียงแค่ปรับอุณหภูมิแต่ไม่สามารถควบคุมอุณหภูมิได้ ดังนั้นในโครงงานนี้จึงนำเสนอการรักษาอุณหภูมิให้เหมาะสมด้วยการออกแบบและสร้างเครื่องควบคุมอุณหภูมิของน้ำภายในตู้ปลา โดยใช้ไมโครคอนโทรเลอร์ทำหน้าที่ประมวลผลจากการเปรียบเทียบค่าอุณหภูมิที่ได้จากไอซี DS18S20 กับค่าที่กำหนดโดยผู้ใช้งาน แล้วสั่งให้พัดลมหรือเครื่องทำความร้อนทำงานเพื่อปรับอุณหภูมิของน้ำให้อยู่ในช่วงที่กำหนด ซึ่งเครื่องควบคุมอุณหภูมิของน้ำที่สร้างขึ้นสามารถใช้กับพัดลมและเครื่องทำความร้อนได้หลายขนาดทำให้ผู้ใช้งานสามารถเลือกขนาดของอุปกรณ์ให้เหมาะสมกับขนาดของตู้ปลา โดยเครื่องควบคุมอุณหภูมิของน้ำภายในตู้ปลาจะช่วยทำให้ปลามีสุขภาพแข็งแรง รูปร่างและสีสันสวยงาม นอกจากนี้ยังสามารถนำไปประยุกต์ใช้กับการรักษาอุณหภูมิในช่วงผสมพันธุ์ปลาหรือใช้กับสัตว์น้ำและพืชไม้น้ำอีกหลายชนิดได้

| | |
|---|---|
| **Project title** | Microcontroller-based Aquarium Temperature Control |
| **Name** | Mr. Nattapon Chobjit ID. 50364546 |
| | Mr. Teerapong Meemanwit ID. 50364607 |
| **Project advisor** | Mr. Niphat Jantharamin, Ph.D. |
| **Major** | Electrical Engineering |
| **Department** | Electrical and Computer Engineering |
| **Academic year** | 2010 |

.....

## Abstract

This thesis presents temperature control of aquarium water. This project arose from the idea that ornamental fish have been popular, both for hobby and commercial purposes. However, Thailand climate may not be suitable for some kinds of imported ornamental fish. Therefore, water temperature should be regulated by means of temperature control devices such as a ventilator for reducing and a heater for raising the temperature. However, the aforementioned devices can only adjust but cannot control the temperature. This project presents control of aquarium water temperature in a desired temperature range. Hereby, a microcontroller reads the water temperature via a DS18S20 and compares it with a user-defined value, and subsequently switches on the ventilator or the heater. The temperature controller developed in this project can be adapted to a wide power range, so a proper size of the ventilators as well as the heaters can be selected for each aquarium. Apart from being useful to the fish health and colour, this controller can be applied to aquatic plants and other species of aquatic animals as well as can support fish breeding.

# กิตติกรรมประกาศ

โครงงานนี้สำเร็จลุล่วงไปได้ด้วยความกรุณาเป็นอย่างยิ่งจาก ดร. นิพัทธ์ จันทรมินทร์ ซึ่งเป็นอาจารย์ที่ปรึกษาโครงงาน และให้ความกรุณาในการตรวจทานปริญญานิพนธ์ คณะผู้ดำเนินโครงงานขอกราบขอบพระคุณเป็นอย่างสูงและขอระลึกถึงความกรุณาของท่านไว้ตลอดไป

นอกจากนี้ยังต้องขอขอบคุณภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ที่ให้ยืมอุปกรณ์และเครื่องมือวัดมาใช้งาน จนทำให้โครงงานนี้สำเร็จลุล่วงไปได้

เหนือสิ่งอื่นใด คณะผู้ดำเนินโครงงานขอกราบขอบพระคุณคุณพ่อ คุณแม่ ผู้มอบความรัก ความเมตตา สติปัญญา รวมทั้งเป็นผู้ให้ทุกสิ่งทุกอย่างตั้งแต่วัยเยาว์จวบจนถึงปัจจุบัน คอยเป็นกำลังใจทำให้ได้รับความสำเร็จอย่างทุกวันนี้ และขอขอบคุณทุก ๆ คนในครอบครัวของคณะผู้ดำเนินโครงงานที่ไม่ได้กล่าวไว้ ณ ที่นี้ด้วย

นายณัฐพล ชอบจิตต์

นายธีรพงศ์ มีแม่นวิทย์

# สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์ ..... ก
บทคัดย่อภาษาไทย ..... ข
บทคัดย่อภาษาอังกฤษ ..... ค
กิตติกรรมประกาศ ..... ง
สารบัญ ..... จ
สารบัญตาราง ..... ช
สารบัญรูป ..... ฌ

บทที่ 1 บทนำ ..... 1
1.1 ที่มาและความสำคัญของโครงงาน ..... 1
1.2 วัตถุประสงค์ของโครงงาน ..... 2
1.3 ขอบเขตของโครงงาน ..... 2
1.4 ขั้นตอนและแผนการดำเนินงาน ..... 2
1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงงาน ..... 3
1.6 งบประมาณ ..... 3

บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง ..... 4
2.1 การเปลี่ยนแปลงอุณหภูมิของน้ำกับปลาสวยงาม ..... 4
2.2 ไมโครคอนโทรลเลอร์ ..... 5
2.2.1 ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ..... 5
2.2.2 ไมโครคอนโทรลเลอร์ หมายเลข P89V51RD2 ..... 6
2.3 ไอซี DS18S20 ..... 7
2.3.1 คำสั่งเพื่อควบคุมการทำงานของ DS18S20 ..... 8
2.3.2 การติดต่อกับ DS18S20 เพื่อรับส่งข้อมูล ..... 9
2.4 รีเลย์ควบคุม ..... 10
2.5 แอลอีดีเจ็ดส่วน ..... 11

# สารบัญ (ต่อ)

หน้า


บทที่ 3 การออกแบบและการทำงานเครื่องควบคุมอุณหภูมิของน้ำ ..... 13
3.1 ส่วนประกอบและวิธีการใช้เครื่องควบคุมอุณหภูมิของน้ำ ..... 13
3.1.1 ส่วนประกอบเครื่องควบคุมอุณหภูมิของน้ำ ..... 13
3.1.2 วิธีการใช้เครื่องควบคุมอุณหภูมิของน้ำ ..... 17
3.2 การทำงานของโปรแกรมเพื่อควบคุมการทำงานของเครื่องควบคุมอุณหภูมิ ..... 18
3.2.1 ขั้นตอนการอ่านค่าอุณหภูมิจากไอซี DS18S20 ..... 18
3.2.2 ขั้นตอนการปรับตั้งค่าอุณหภูมิ ..... 19
3.2.3 ขั้นตอนควบคุมการทำงานรีเลย์ ..... 22
3.3 วงจรไมโครคอนโทรลเลอร์ MCS-51 ..... 23
3.3.1 ส่วนวงจรควบคุมแรงดัน ..... 24
3.3.2 ส่วนวงจรป้อนข้อมูล ..... 24
3.3.3 ส่วนวงจรไมโครคอนโทรลเลอร์ ..... 25
3.4 วงจรแอลอีดีเจ็ดส่วน ..... 27
3.5 วงจรควบคุมการทำงานอุปกรณ์ปรับอุณหภูมิ ..... 28
3.6 วงจรไอซี DS18S20 ..... 29

บทที่ 4 ผลการทดสอบ ..... 30
4.1 การทดสอบการอ่านค่าอุณหภูมิของไอซี DS18S20 ..... 30
4.2 การทดสอบความสามารถในการปรับอุณหภูมิ ..... 32

บทที่ 5 สรุปผลและข้อเสนอแนะ ..... 39
5.1 สรุปผลการดำเนินโครงงาน ..... 39
5.2 ปัญหาและแนวทางแก้ไข ..... 39
5.3 แนวทางในการพัฒนาต่อไป ..... 40

เอกสารอ้างอิง ..... 41

# สารบัญ (ต่อ)

หน้า

ภาคผนวก ก โปรแกรมควบคุมเครื่องควบคุมอุณหภูมิน้ำ ..... 42
ภาคผนวก ข รายละเอียดของไอซีหมายเลข DS18S20 ..... 54
ภาคผนวก ค รายละเอียดของทรานซิสเตอร์หมายเลข BC327 ..... 59
ภาคผนวก ง รายละเอียดของทรานซิสเตอร์หมายเลข BC337 ..... 63

ประวัติผู้ดำเนินโครงงาน ..... 67

# สารบัญตาราง

ตารางที่ หน้า

1.1 อุณหภูมิเฉลี่ยสูงสุดและต่ำสุด พ.ศ. 2553 ประจำวันของประเทศไทย [1] ..... 1
3.1 ข้อมูล 2 บิต ที่ได้จากไอซี DS18S20 ..... 19
3.2 พอร์ตอินพุตของไมโครคอนโทรลเลอร์ ..... 26
3.3 พอร์ตเอาท์พุตของไมโครคอนโทรลเลอร์ ..... 27
4.1 ผลการทดสอบระดับอุณหภูมิที่ต่ำกว่าอุณหภูมิน้ำ 2 องศา ..... 33
4.2 ผลการทดสอบระดับอุณหภูมิที่มากกว่าอุณหภูมิน้ำ 2 องศา ..... 34

# สารบัญรูป

รูปที่ หน้า


2.1 วงจรไมโครคอนโทรลเลอร์ P89V51RD2 [3] ..... 6
2.2 แสดงการจัดขาของไอซี DS18S20 ..... 7
2.3 โครงสร้างการทำงานภายในของไอซี DS18S20 ..... 7
2.4 การจัดสรรพื้นที่ของสแครตซ์แพดใน DS18S20 ..... 8
2.5 สภาวะการทำงานของรีเลย์ ..... 10
2.6 ลักษณะภายนอกของรีเลย์ ..... 11
2.7 การต่อแอลอีดีเจ็ดส่วนแบบแอโนดร่วม ..... 11
2.8 การต่อแอลอีดีเจ็ดส่วนแบบแคโทดร่วม ..... 12
2.9 แอลอีดีเจ็ดส่วนแบบตัวเดี่ยวและแบบตัวคู่ ..... 12
3.1 การทำงานเครื่องควบคุมอุณหภูมิของน้ำ ..... 14
3.2 ส่วนประกอบภายนอกของเครื่องควบคุมอุณหภูมิของน้ำ ..... 15
3.3 ส่วนประกอบภายในของเครื่องควบคุมอุณหภูมิของน้ำ ..... 16
3.4 แผนภาพการทำงานของเครื่องควบคุมอุณหภูมิ ..... 18
3.5 แผนภาพการทำงานโหมดการปรับตั้งค่าอุณหภูมิ ..... 20
3.6 แผนภาพการทำงานโหมดปรับตั้งช่วงอุณหภูมิที่ต้องการรักษา ..... 21
3.7 แผนภาพการทำงานขั้นตอนการควบคุมการทำงานรีเลย์ ..... 23
3.8 แผนภาพกรอบการทำงานในส่วนวงจรควบคุมแรงดัน ..... 24
3.9 วงจรควบคุมแรงดัน ..... 24
3.10 แผนภาพกรอบการทำงานในส่วนวงจรป้อนข้อมูล ..... 25
3.11 วงจรป้อนข้อมูล ..... 25
3.12 วงจรไมโครคอนโทรลเลอร์ [5] ..... 26
3.13 วงจรแอลอีดีเจ็ดส่วน ..... 27
3.14 วงจรควบคุมการทำงานอุปกรณ์ปรับอุณหภูมิ ..... 29
3.15 วงจรDS18S20 ..... 29
4.1 ผลการทดลองการอ่านค่าอุณหภูมิของไอซี DS18S20 ที่ช่วงอุณหภูมิเพิ่มขึ้น ..... 31
4.2 ผลการทดลองการอ่านค่าอุณหภูมิของไอซี DS18S20 ที่ช่วงอุณหภูมิลดลง ..... 31

# สารบัญรูป (ต่อ)

รูปที่ หน้า

4.3 กราฟความสัมพันธ์ระหว่างระดับอุณหภูมิที่ลดลงกับเวลา ..... 36
4.4 กราฟความสัมพันธ์ระหว่างระดับอุณหภูมิที่เพิ่มขึ้นกับเวลา ..... 37

# บทที่ 1
# บทนำ

## 1.1 ที่มาและความสำคัญของโครงงาน

การเลี้ยงปลาสวยงามในปัจจุบันเริ่มเป็นที่นิยมแพร่หลายมากขึ้นในประเทศไทยทั้งการเลี้ยงเป็นงานอดิเรกหรือในเชิงเศรษฐกิจ แต่ปัญหาที่มักพบกันมากในกลุ่มผู้เลี้ยงปลาสวยงามคือ ปลาสวยงามมีรูปร่างและสีสันไม่สวยงาม ไม่ค่อยกินอาหาร และเป็นโรค ปัญหาเหล่านี้เกิดได้หลายสาเหตุ เช่น น้ำไม่สะอาด ออกซิเจนในน้ำไม่เพียงพอ การอยู่อย่างแออัดของปลา เป็นต้น และอีกหนึ่งสาเหตุสำคัญคือ อุณหภูมิของน้ำไม่เหมาะสมกับปลาซึ่งเป็นปัญหาที่พบมากในประเทศไทยซึ่งมีการนำเข้าปลาสวยงามจากต่างประเทศมากมายหลายสายพันธุ์ ประเทศไทยมีภูมิอากาศแบบเขตร้อน มีอุณหภูมิเฉลี่ยตลอดปีประมาณ 18-34 องศาเซลเซียส อุณหภูมิเฉลี่ยสูงสุดและต่ำสุดประจำวันแสดงดังตารางที่ 1.1

ตารางที่ 1.1 อุณหภูมิเฉลี่ยสูงสุดและต่ำสุด พ.ศ. 2553 ประจำวันของประเทศไทย [1]

| ฤดูกาล / ภูมิภาค | ฤดูร้อน มี.ค. – มิ.ย. | ฤดูฝน ก.ค. – ต.ค. | ฤดูหนาว พ.ย. – ก.พ. |
|---|---|---|---|
| ภาคกลาง | 30/20°C | 32/28°C | 30/20°C |
| ภาคเหนือ | 44/46°C | 28/20°C | 28/24°C |
| ภาคใต้ | 32/28°C | 30/20°C | 44/46°C |
| ภาคตะวันออก | 28/20°C | 44/46°C | 32/28°C |
| ภาคตะวันตก | 28/24°C | 30/20°C | 28/20°C |
| ภาคตะวันออกเฉียงเหนือ | 30/55°C | 44/46°C | 32/28°C |

อุณหภูมิที่แตกต่างกันตามฤดูของประเทศไทยทำให้ไม่เหมาะกับปลาสวยงามที่ทนต่อการเปลี่ยนแปลงสภาพอากาศได้น้อย โดยทั่วไปปลาในเขตร้อนมักจะชอบอาศัยอยู่ในอุณหภูมิระหว่าง 25–32 องศาเซลเซียส แต่กับปลาสวยงามการเปลี่ยนแปลงอุณหภูมิเพียง 2–5 องศาเซลเซียส ก็สามารถทำให้ปลาตายหรือก่อให้เกิดสภาพอ่อนแอได้ แน่นอนว่าการเลี้ยงปลาสวยงามถูกเลี้ยงอยู่ในสถานที่แคบ ๆ การเปลี่ยนแปลงอุณหภูมิของน้ำจึงเปลี่ยนแปลงได้ง่ายตามสภาพแวดล้อมรอบข้าง ทำให้การควบคุมอุณหภูมิให้คงที่เป็นไปได้ยาก

ปัจจุบันมีอุปกรณ์เพิ่มหรือลดอุณหภูมิ เช่น พัดลมระบายอากาศและเครื่องทำความร้อน ที่วางขายอยู่ในตลาดปลาสวยงาม แม้อุปกรณ์เหล่านั้นจะทำหน้าที่ปรับอุณหภูมิแต่ไม่ได้ทำหน้าที่ควบคุมอุณหภูมิ ดังนั้นผู้ดำเนินโครงงานจึงได้สร้างเครื่องควบคุมอุณหภูมิของน้ำขึ้นให้อุณหภูมิอยู่ในช่วงที่ต้องการซึ่งเป็นการช่วยให้ปลามีสุขภาพแข็งแรง รูปร่างและสีสันสวยงาม

## 1.2 วัตถุประสงค์ของโครงงาน

เพื่อรักษาอุณหภูมิของน้ำในตู้ปลาให้อยู่ในช่วงอุณหภูมิที่เหมาะสมกับปลาสวยงาม

## 1.3 ขอบเขตของโครงงาน

1) สร้างเครื่องควบคุมอุณหภูมิของน้ำที่มีค่าความคลาดเคลื่อนในการอ่านอุณหภูมิไม่เกิน 0.5 องศาเซลเซียส ที่ตู้ปลาขนาดไม่เกิน 24 นิ้วหรือปริมาณน้ำไม่เกิน 54 ลิตร
2) เครื่องควบคุมอุณหภูมิของน้ำที่สร้างขึ้นสามารถปรับอุณหภูมิให้อยู่ในขอบเขตการควบคุมที่ผู้ใช้งานกำหนดไว้ได้

## 1.4 ขั้นตอนและแผนการดำเนินงาน

| รายละเอียด | ปี 2553 | | | | | | | ปี 2554 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | มิ.ย. | ก.ค. | ส.ค. | ก.ย. | ต.ค. | พ.ย. | ธ.ค. | ม.ค. | ก.พ. | มี.ค. |
| 1) ศึกษาเรื่องอุณหภูมิของน้ำและระบบนิเวศน์ปลา | ■ | | | | | | | | | |
| 2) ศึกษาเรื่องระบบไมโครคอนโทรลเลอร์ | | ■ | ■ | | | | | | | |
| 3) สร้างเครื่องควบคุมอุณหภูมิของน้ำ | | | ■ | ■ | ■ | ■ | | | | |
| 4) ทดสอบและปรับปรุงระบบให้สมบูรณ์ | | | | | | ■ | ■ | ■ | | |
| 5) สรุปผลการดำเนินงานและจัดทำปริญญานิพนธ์ | | | | | | | | | ■ | ■ |

## 1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงงาน

การรักษาระดับอุณหภูมิให้อยู่ในช่วงที่เหมาะสมกับปลาสวยงามช่วยลดปัญหาต่าง ๆ ที่เกิดจากระดับอุณหภูมิไม่เหมาะสมกับปลาสวยงามได้ เช่น มีรูปร่างและสีสันไม่สวยงาม ไม่ค่อยกินอาหาร เกิดโรค เป็นต้น

## 1.6 งบประมาณ

| | |
|---|---|
| 1) ค่าใช้จ่ายในวงจรควบคุม | 400 บาท |
| 2) ค่าใช้จ่ายในวงจรรีเลย์ | 100 บาท |
| 3) เครื่องทำความร้อน | 300 บาท |
| 4) พัดลมระบายอากาศ | 400 บาท |
| 5) ค่าใช้จ่ายในส่วนอื่น ๆ | 200 บาท |
| 6) ค่าถ่ายเอกสารและเข้าเล่มปริญญานิพนธ์ | 800 บาท |
| รวมเป็นเงินทั้งสิ้น (สองพันหนึ่งร้อยบาทถ้วน) | 2,200 บาท |

หมายเหตุ: ถัวเฉลี่ยทุกรายการ

# บทที่ 2
# ทฤษฎีและหลักการที่เกี่ยวข้อง

การทำงานเครื่องควบคุมอุณหภูมิของน้ำเป็นการนำเอาความสามารถของไมโครคอนโทรลเลอร์ตระกูล MCS-51 มาเป็นหัวใจหลักในการควบคุมการทำงานของเครื่องควบคุมอุณหภูมิของน้ำ ในบทนี้จึงได้เสนอหลักการทำงานของไมโครคอนโทรลเลอร์ พร้อมทั้งหลักการของภาษาแอสเซมบลีซึ่งเป็นภาษาที่เลือกมาใช้เขียนโปรแกรมเพื่อควบคุมการทำงาน และยังนำเอาหลักการทำงานของอุปกรณ์อิเล็กทรอนิกส์สำคัญที่ประกอบเป็นวงจรอยู่ในเครื่องควบคุมอุณหภูมิของน้ำมานำเสนอเพื่อความเข้าใจในการทำงานของเครื่องมากขึ้น และเพื่อความเข้าใจความสำคัญของการควบคุมอุณหภูมิของน้ำโครงงานนี้ ผู้ดำเนินโครงงานจึงได้หยิบยกเรื่องการเปลี่ยนแปลงอุณหภูมิของน้ำกับปลาสวยงามมานำเสนอเป็นหัวข้อแรก

## 2.1 การเปลี่ยนแปลงอุณหภูมิของน้ำกับปลาสวยงาม

ปลาในเขตร้อนมักจะชอบอาศัยอยู่ในอุณหภูมิระหว่าง 25–32 องศาเซลเซียส อุณหภูมิมีส่วนสำคัญต่อขบวนการต่าง ๆ ทั้งด้านเคมีและชีววิทยาภายในร่างกายของสิ่งมีชีวิต เมื่ออุณหภูมิสูงขึ้นแต่ไม่เกินขอบเขตที่กำหนดปลาจะหายใจได้ดีขึ้น และขบวนการอื่นภายในร่างกาย เช่น การย่อยอาหาร การหายใจ การเจริญเติบโตจะสูงขึ้นด้วย เป็นต้น นอกจากนี้การใช้สารเคมีในแหล่งน้ำที่มีอุณหภูมิสูงกว่าระดับปกติก็จะทำให้ปฏิกิริยาต่าง ๆ เกิดขึ้นอย่างรวดเร็ว ตัวอย่างเช่น ยากำจัดศัตรูพืชจะทำงานรวดเร็วและสลายตัวเร็วขึ้น ตลอดจนความต้องการออกซิเจนในการย่อยสลายสารอินทรีย์ต่าง ๆ ของแบคทีเรียจะเพิ่มมากขึ้นด้วย ในขณะเดียวกับที่น้ำลดประสิทธิภาพในการรับออกซิเจนลง จึงทำให้เกิดปัญหาการขาดแคลนออกซิเจนขึ้นได้ แหล่งน้ำของประเทศในเขตหนาวจะเกิดการแบ่งชั้นของอุณหภูมิในระดับต่าง ๆ ตามความลึก แต่ในเขตร้อนความแตกต่างของอุณหภูมิน้ำระหว่างชั้นบนกับชั้นล่างมักจะไม่เห็นเด่นชัด โดยเฉพาะในบ่อตื้น ๆ ซึ่งในตอนกลางวันผิวน้ำจะมีอุณหภูมิสูงกว่าก้นบ่อและแบ่งเป็นชั้น แต่ในช่วงกลางคืนอุณหภูมิผิวน้ำจะลดลงเท่ากับอุณหภูมิชั้นล่างทำให้น้ำรวมตัวกันได้ ในบางกรณีอุณหภูมิของน้ำชั้นบนจะมากกว่า 35 องศาเซลเซียส ซึ่งไม่เหมาะกับปลาบางชนิด ดังนั้นในการเลี้ยงปลาสวยงามจึงไม่ควรให้อุณหภูมิเปลี่ยนแปลงอย่างรวดเร็ว การเปลี่ยนแปลงของอุณหภูมิเพียง 2–5 องศาเซลเซียส สามารถทำให้ปลาตายได้โดยเฉพาะเมื่อนำปลาจากที่อุณหภูมิต่ำกว่าไปยังที่อุณหภูมิสูงกว่าจะเกิดผลร้ายแรงต่อปลากว่าการนำปลาจากที่อุณหภูมิสูงไปยังที่อุณหภูมิต่ำ ดังนั้นในการเคลื่อนย้ายปลาจึงต้องค่อย ๆ ให้ปลาเจอกับการเปลี่ยนแปลงอุณหภูมิทีละน้อยอย่างช้า ๆ เพื่อให้ปลาสามารถปรับตัว

ได้ทัน และอุณหภูมิของน้ำถือเป็นปัจจัยสำคัญที่จะทำให้ปลาอยากหรือเบื่ออาหาร ถ้าอุณหภูมิของน้ำเหมาะสมกับปลาจะทำให้ปลามีความอยากอาหารมากขึ้นส่งผลให้ปลามีการเจริญเติบโตดี แต่ถ้าหากอุณหภูมิลดลงต่ำกว่าขีดที่เหมาะสมแล้ว ปลาอาจจะหยุดกินอาหารหรือกินน้อยลง [2]

การศึกษาของผลของการเปลี่ยนแปลงอุณหภูมิของน้ำที่มีต่อปลาสวยงาม ทำให้รู้ถึงความจำเป็นในการควบคุมอุณหภูมิของน้ำให้อยู่ในช่วงที่ปลาต้องการ และในส่วนต่อไปเป็นการศึกษาเกี่ยวกับหลักการทำงานของอุปกรณ์ภายในเครื่องควบคุมอุณหภูมิของน้ำ

## 2.2 ไมโครคอนโทรลเลอร์

โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์นั้น สามารถแบ่งออกมาได้เป็น 5 ส่วนใหญ่ ๆ ดังต่อไปนี้

1) หน่วยประมวลผลกลางหรือซีพียู (Central Processing Unit: CPU)
2) หน่วยความจำ (Memory)
3) ส่วนติดต่อกับอุปกรณ์ภายนอก หรือพอร์ต (Port)
4) ช่องทางเดินของสัญญาณ หรือบัส (Bus)
5) วงจรกำเนิดสัญญาณนาฬิกา

### 2.2.1 ไมโครคอนโทรลเลอร์ตระกูล MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 มีการพัฒนาให้มีความสามารถและมีความเร็วเพิ่มขึ้น แต่ยังคงโครงสร้างพื้นฐานที่สำคัญของไมโครคอนโทรลเลอร์ตระกูล 8051 โดยจะเป็นไมโครคอนโทรลเลอร์ตระกูลเริ่มแรกก่อน MCS-51 ซึ่งมีรายละเอียดดังนี้

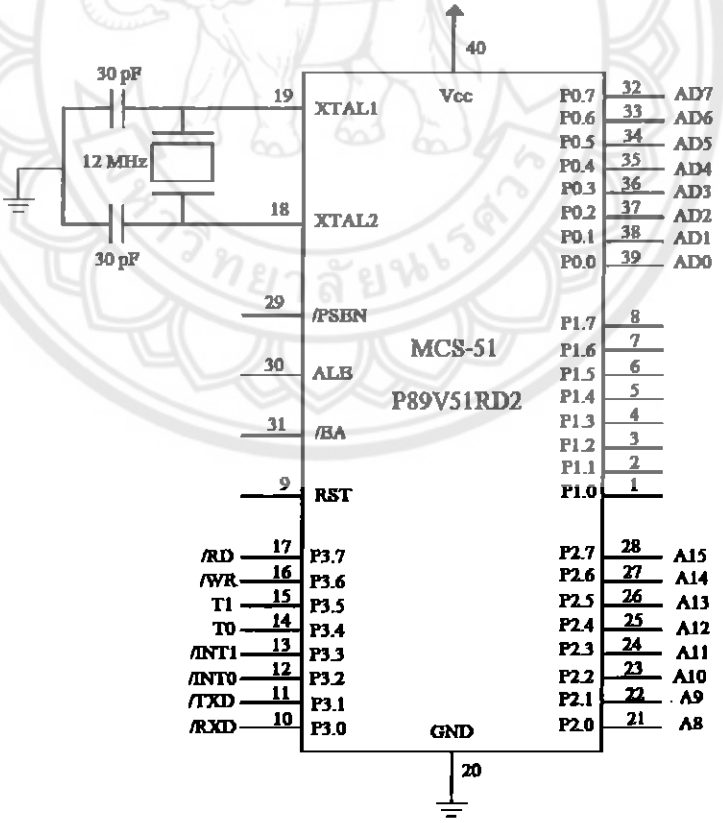
1) เป็นไมโครคอนโทรลเลอร์ที่มีหน่วยประมวลผลกลางแบบ 8 บิต
2) มีคำสั่งคำนวณทางคณิตศาสตร์ และตรรกศาสตร์ (Boolean processor)
3) มีแอดเดรสบัสขนาด 16 บิตทำให้สามารถอ้างตำแหน่งหน่วยความจำโปรแกรม และหน่วยความจำข้อมูลได้ 64 กิโลไบต์
4) มีหน่วยความจำชั่วคราว (RAM) ภายในขนาด 128 ไบต์ หรือ 256 ไบต์
5) มีพอร์ตอนุกรมทำงานแบบดูเพล็กซ์เต็ม (Full duplex) 1 พอร์ต
6) มีพอร์ตอินพุต/เอาท์พุตแบบขนานจำนวน 32 บิต
7) มีตัวจับเวลา 2 ตัว (8051/8031) หรือ 3 ตัว (8052/8032)
8) มีวงจรควบคุมการเกิดอินเตอร์รัพท์ 5 ประเภท หรือ 6 ประเภท
9) มีวงจรออสซิลเลเตอร์

ไมโครคอนโทรลเลอร์ MCS-51 มีวงจรออสซิลเลเตอร์อยู่ภายใน ดังนั้นในการใช้งานจึงสามารถต่อคริสตอล (Crystal) และตัวเก็บประจุเข้ากับคริสตอลได้โดยตรง โดยความถี่ของคริสตอลที่ต่อเข้ากับไมโครคอนโทรลเลอร์จะเป็นตัวระบุความเร็วในการทำงานโดยตรง ในไมโครคอนโทรลเลอร์ MCS-51 ปกติ 1 แมชชีนไซเคิล (Machine cycle) จะใช้สัญญาณนาฬิกาจำนวน 12 ลูก และในการทำงานแต่ละคำสั่งไมโครคอนโทรลเลอร์จะใช้เวลาในการทำงาน 1-4 แมชชีนไซเคิล ขึ้นอยู่กับความซับซ้อนของคำสั่งนั้น

ในปัจจุบันผู้ผลิตได้พัฒนาให้ไมโครคอนโทรลเลอร์สามารถทำงานได้เร็วขึ้นโดยเพิ่มความสามารถในการรองรับคริสตอลความถี่ที่สูงขึ้น รวมไปถึงการปรับปรุงการทำงานภายในให้ไมโครคอนโทรลเลอร์ใช้จำนวนสัญญาณนาฬิกาในการสร้างแมชชีนไซเคิลน้อยลง โดยในบางรุ่น 1 แมชชีนไซเคิลใช้สัญญาณนาฬิกาเพียงแค่ 1 ลูกเท่านั้น

### 2.2.2 ไมโครคอนโทรลเลอร์ หมายเลข P89V51RD2

ไมโครคอนโทรลเลอร์ หมายเลข P89V51RD2 เป็นไมโครคอนโทรลเลอร์หมายเลขหนึ่งในตระกูล MCS-51 มีลักษณะโครงสร้างภายนอกและการต่อวงจรดังรูปที่ 2.1

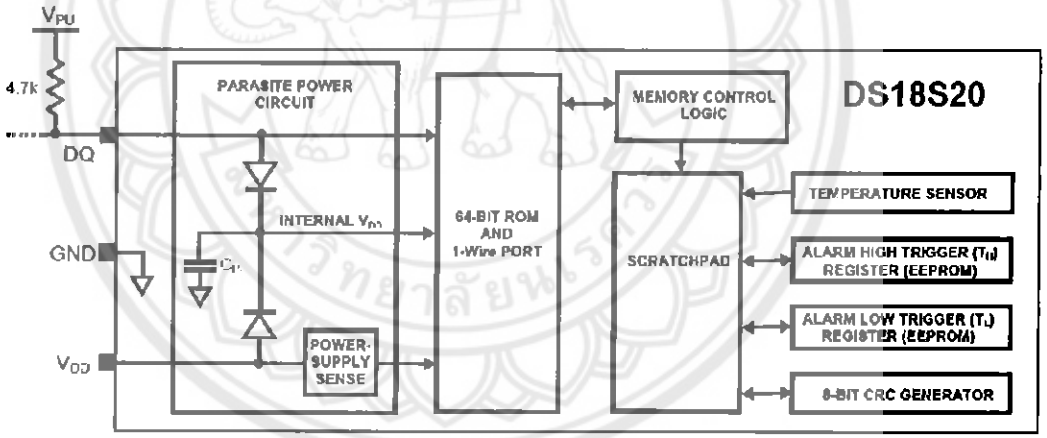



**รูปที่ 2.1** วงจรไมโครคอนโทรลเลอร์ P89V51RD2 [3]

## 2.3 ไอซี DS18S20

DS18S20 เป็นไอซีตรวจจับอุณหภูมิที่ใช้การติดต่อแบบระบบบัสหนึ่งสาย มีขาต่อใช้งาน 3 ขา คือ DQ เป็นขาเชื่อมต่อระบบบัส, ขา VDD ต่อไฟเลี้ยงภายนอก และขา GND ดังแสดงการจัดขาของไอซี DS18S20 ในรูปที่ 2.2 และมีโครงสร้างภายในประกอบด้วยหน่วยความจำภายใน 64 บิต และหน่วยความจำชั่วคราวอีก 9 ไบต์ ที่เรียกว่าสแครตช์แพด (Scratchpad) ดังรูปที่ 2.3

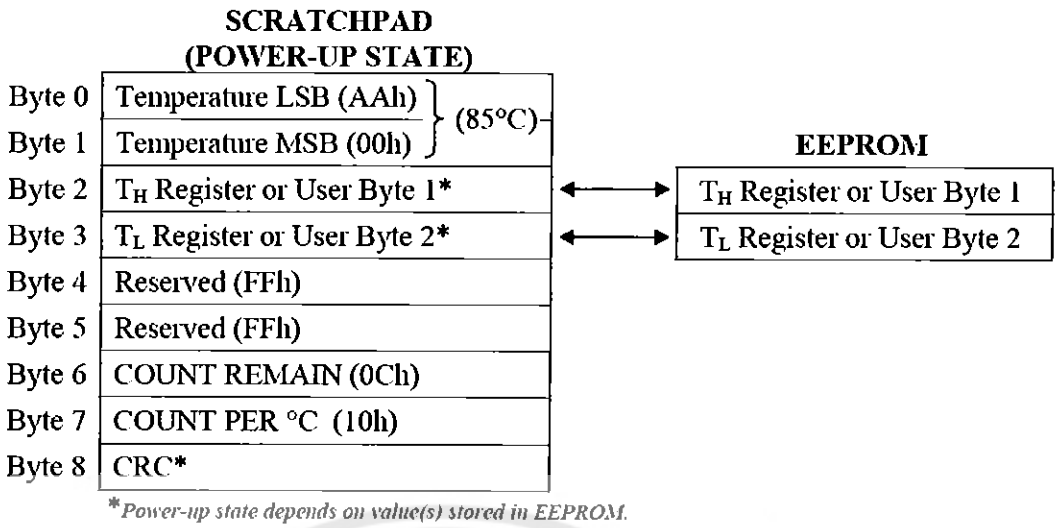
รูปที่ 2.2 แสดงการจัดขาของไอซี DS18S20




รูปที่ 2.3 โครงสร้างการทำงานภายในของไอซี DS18S20

ที่มา: DALLAS SEMICONDUCTOR (2554)

หัวใจสำคัญของ DS18S20 อยู่ที่ตัวตรวจจับอุณหภูมิและหน่วยความจำความเร็วสูงที่เรียกว่าสแครตช์แพด มีการจัดสรรหน่วยความจำส่วนนี้แสดงในรูปที่ 2.4




**รูปที่ 2.4** การจัดสรรพื้นที่ของสแครตช์แพดใน DS18S20

เมื่อวัดอุณหภูมิได้ก็จะนำค่าที่วัดได้นี้มาเก็บไว้ในสแครตช์แพดที่ไบต์ 0 และ 1 ทั้งนี้เนื่องจากไอซี DS18S20 สามารถให้ข้อมูลของอุณหภูมิได้ละเอียดถึง 16 บิตเมื่อนำมาแปลงเป็นเลขฐานสิบจึงสามารถแสดงความละเอียดของอุณหภูมิได้ละเอียดถึง 0.5 องศาเซลเซียส และ 0.9 องศาฟาเรนไฮต์ โดยมีย่านวัดอุณหภูมิ -55 ถึง +125 องศาเซลเซียสหรือ -67 ถึง +257 องศาฟาเรนไฮต์ โดยค่าขององศาฟาเรนไฮต์ต้องใช้การแปลงหน่วยเข้ามาช่วยใช้เวลาในการแปลงค่าอุณหภูมิเป็นข้อมูลดิจิตอลประมาณ 200 มิลลิวินาที สามารถกำหนดขอบเขตของอุณหภูมิที่ทำการวัดได้ และให้แจ้งเตือนเมื่อค่าอุณหภูมิสูงขึ้นหรือลดต่ำลงถึงค่าที่กำหนด โดยค่าอุณหภูมิที่กำหนดนี้จะเก็บไว้ที่สแครตช์แพดในไบต์ 2 และ 3

### 2.3.1 คำสั่งเพื่อควบคุมการทำงานของ DS18S20

ในการติดต่อกับไอซี DS18S20 จะมีคำสั่งที่ต้องส่งให้แก่ DS18S20 เพื่อกำหนดรูปแบบการทำงานคำสั่งที่ใช้มากที่สุดมีด้วยกัน 3 คำสั่งดังนี้

1) **คำสั่งไม่ติดต่อกับหน่วยความจำรอมหรือสคิปรอม (Skip ROM)** เนื่องจากในการใช้งานโดยส่วนใหญ่มักจะมี DS18S20 อยู่บนสายสัญญาณเพียงตัวเดียว จึงไม่จำเป็นต้องใช้ข้อมูลกำหนดแอดเดรส ดังนั้นจึงไม่ต้องติดต่อกับหน่วยความจำรอมเพื่ออ่านข้อมูล ข้อมูลของคำสั่งสคิปรอมที่ต้องส่งให้ DS18S20 คือ 0CCH
2) **คำสั่งแปลงอุณหภูมิ (Convert T)** มีค่าเท่ากับ 44H เมื่อส่งคำสั่งนี้ให้ DS18S20 จะต้องทำการวนลูปรออย่างน้อย 200 มิลลิวินาที เพื่อให้ DS18S20 ได้ใช้เวลานี้ในการแปลงอุณหภูมิเป็นข้อมูลดิจิตอลมาเก็บไว้ในสแครตช์แพด

**3) คำสั่งอ่านข้อมูลจากสแครตซ์แพด (Read scratchpad)** มีค่าเท่ากับ 0BEH เมื่อส่งคำสั่งนี้ DS18S20 จะทยอยส่งข้อมูลค่าอุณหภูมิออกมาทั้งหมด 9 ไบต์

**2.3.2 การติดต่อกับ DS18S20 เพื่อรับส่งข้อมูล**

เนื่องจาก DS18S20 มีการรับส่งข้อมูลแบบหนึ่งสาย (1-Wire serial bus) คือสายสัญญาณบนระบบบัสแบบหนึ่งสายจะเป็นสายสัญญาณแบบสองทิศทาง ดังนั้นการติดต่อกับ DS18S20 จึงมีการติดต่อสื่อสารแบบหนึ่งสายดังนี้

**1) ระบบสื่อสารข้อมูลอนุกรมแบบหนึ่งสาย**

ระบบการสื่อสารข้อมูลแบบนี้ผู้ค้นคิดคือ ดัลลัสเซมิคอนดักเตอร์ ดังนั้นในบางครั้งจึงเรียกระบบสื่อสารข้อมูลแบบนี้ว่า ระบบสื่อสารข้อมูลดัลลัสหนึ่งสาย (The dallas 1-wire bus) ระบบสื่อสารข้อมูลแบบนี้เป็นระบบที่มีความชาญฉลาด และใช้จำนวนสายสัญญาณเพียง 1 เส้นเท่านั้น โดยไม่ต้องมีสายสัญญาณนาฬิกามาควบคุมจังหวะการถ่ายทอดข้อมูลเหมือนกับระบบสื่อสารข้อมูลอนุกรมในแบบอื่น ๆ เนื่องจากสายข้อมูลนั้นมีหน้าที่เสมือนหนึ่งเป็นสายสัญญาณนาฬิกาในตัว ส่วนค่าของข้อมูลจะพิจารณาจากลักษณะของรูปสัญญาณจะปรากฏบนสายสัญญาณในแต่ละช่องของเวลาหรือต่อไปนี้จะขอเรียกว่า ไทม์สล็อต (Time-slot) ซึ่งโดยคาบเวลาต่ำสุดและสูงสุดของสถานะต่าง ๆ ในการสื่อสารข้อมูลในแต่ละไทม์สล็อต มีการกำหนดขอบเขตไว้อย่างชัดเจนการถ่ายทอดข้อมูลจะเกิดขึ้นในแต่ละไทม์สล็อตนั้น รูปแบบการถ่ายทอดข้อมูลจะเป็นแบบอะซิงโครนัสในระดับบิต ไม่มีการกำหนดความยาวของข้อมูลเป็นระดับไบต์ ระบบสื่อสารแบบนี้เหมาะที่จะใช้ในการสื่อสารข้อมูลระหว่างไอซีบนแผงวงจรเดียวกัน

2) คุณสมบัติทางเทคนิคของระบบบัสหนึ่งสาย

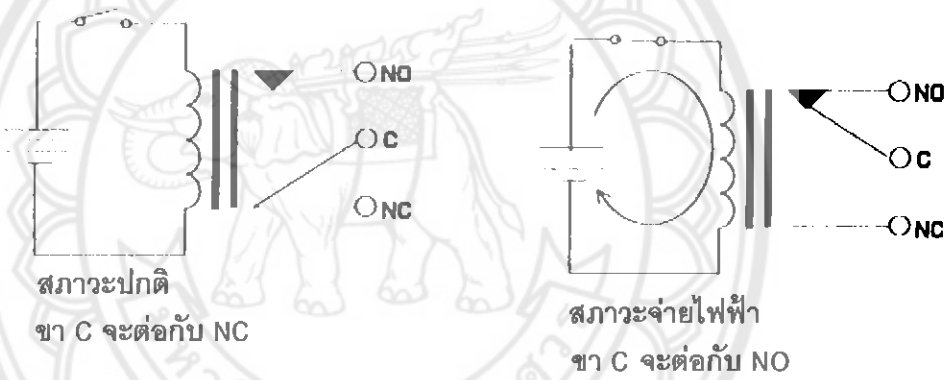
สายสัญญาณบนระบบบัสแบบหนึ่งสายนี้จะเป็นสายสัญญาณแบบสองทิศทาง แต่ข้อมูลจะสามารถเดินทางได้ในทิศทางเดียวภายในช่วงเวลาหนึ่ง ๆ นั่นคือ มีลักษณะคล้ายกับระบบสื่อสารแบบฮาล์ฟดูเฟล็กซ์ (Half duplex) ตัวอย่างที่เห็นได้ชัดคือ การใช้งานวิทยุสื่อสารหรือวิทยุสมัครเล่น อุปกรณ์บนระบบบัสต้องมีการระบุอย่างชัดเจนว่า ตัวใดเป็นอุปกรณ์มาสเตอร์ ตัวใดเป็นอุปกรณ์สเลฟ ส่วนใหญ่อุปกรณ์มาสเตอร์คือไมโครคอนโทรลเลอร์ ส่วนอุปกรณ์สเลฟได้แก่ ไอซีตรวจจับอุณหภูมิ, หน่วยความจำแรม เป็นต้น อุปกรณ์มาสเตอร์จะเป็นตัวจัดเตรียมความพร้อมของสายสัญญาณและควบคุมการถ่ายทอดข้อมูลบนสายสัญญาณนั้น ข้อมูลทั้งหมดไม่ว่าจะเป็นข้อมูลควบคุมหรือข้อมูลใช้งานจะถูกส่งลงบนสายสัญญาณเส้นเดียวนี้ทั้งหมด ในระหว่างการทำงานอุปกรณ์มาสเตอร์และสเลฟสามารถเป็นได้ทั้งตัวส่งและตัวรับ ขึ้นอยู่กับเงื่อนไขของการทำงานในขณะนั้น ยกตัวอย่าง ถ้ามีการเขียนข้อมูลจากอุปกรณ์มาสเตอร์ไปยังอุปกรณ์

สเลฟ ตัวส่งคืออุปกรณ์มาสเตอร์ ตัวรับคืออุปกรณ์สเลฟ ในทางตรงข้าม หากเป็นการอ่านข้อมูลจากอุปกรณ์สเลฟ ตัวส่งจะกลายเป็นอุปกรณ์สเลฟ และตัวรับคืออุปกรณ์มาสเตอร์ ในระบบบัส 1 ระบบต้องมีอุปกรณ์มาสเตอร์เพียงตัวเดียวเท่านั้น

## 2.4 รีเลย์ควบคุม

รีเลย์ควบคุม (Control relay) เป็นอุปกรณ์อิเล็กทรอนิกส์เชิงกลชนิดหนึ่งซึ่งทำหน้าที่เป็นสวิตช์เปิดปิดวงจร แต่การทำงานของรีเลย์นั้นถูกควบคุมด้วยกระแสไฟฟ้า

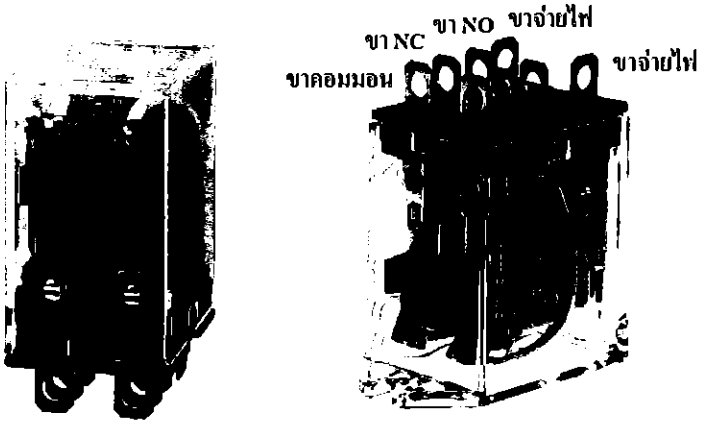
การทำงานของรีเลย์ คือ เมื่อมีการปล่อยกระแสไฟฟ้าให้ไหลผ่านขดลวด จะทำให้ขดลวดเกิดสนามแม่เหล็กไปดึงแผ่นหน้าสัมผัสให้ดึงลงมาแตะหน้าสัมผัสอีกอันทำให้มีกระแสไหลผ่านหน้าสัมผัสไปได้ [4]




รูปที่ 2.5 สภาวะการทำงานของรีเลย์

จากรูปที่ 2.5 แสดงการทำงานของรีเลย์โดยภายในรีเลย์จะประกอบไปด้วยตำแหน่งหน้าสัมผัสต่าง ๆ ดังนี้

1) ขา NO (Normally opened หรือ ปกติเปิด) โดยปกติขานี้จะเปิดเอาไว้ จะทำงานเมื่อเราป้อนแรงดันให้รีเลย์
2) ขา NC (Normally closed หรือ ปกติปิด) โดยปกติขานี้จะต่อกับขา C ในกรณีที่เราไม่ได้จ่ายแรงดัน หน้าสัมผัสของ C และ NC จะต่อถึงกัน
3) ขาคอมมอนหรือ C (COM) จะเป็นขาต่อระหว่าง NO และ NC



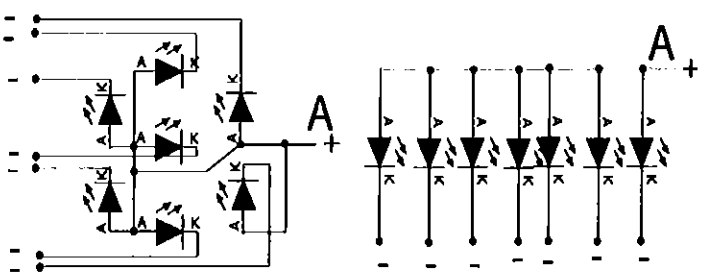

รูปที่ 2.6 ลักษณะภายนอกของรีเลย์

จากรูปที่ 2.6 ที่ขาจ่ายไฟมีอยู่ด้วยกัน 2 ขา ซึ่งการจ่ายไฟเข้าที่ขดลวดรีเลย์ควบคุมสามารถจ่ายไฟเข้าได้ทั้ง 2 ทาง

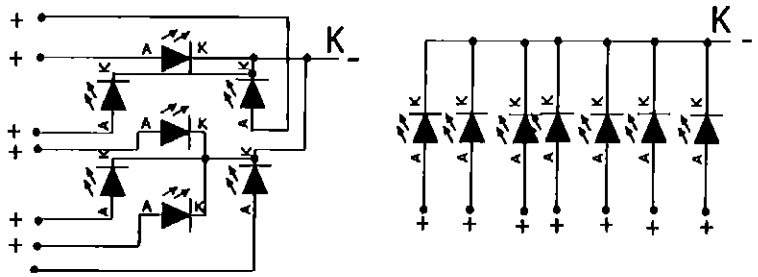
## 2.5 แอลอีดีเจ็ดส่วน

แอลอีดีเจ็ดส่วน (Seven segment LED) ประกอบขึ้นจาก แอลอีดีจำนวน 7 ตัวที่บรรจุอยู่ในตัวถังเดียวกันและได้รับการจัดเรียงเป็นรูปตัวเลข แอลอีดีแต่ละตัวจะถูกเรียกว่า ส่วน (Segment) แต่ละส่วนมีชื่อเรียกแตกต่างกันตามตำแหน่งที่ได้รับการจัดวางคือ a, b, c, d, e, f และ g ส่วน dp เป็นแอลอีดีอีก 1 ตัวที่บรรจุอยู่ในแอลอีดีเจ็ดส่วนนี้ใช้เป็นตัวแสดงจุดทศนิยมในกรณีที่มีการแสดงผลในลักษณะเลขที่มีทศนิยม [5]

แอลอีดีทุกตัวที่บรรจุอยู่ในแอลอีดีเจ็ดส่วนนี้มีขาต่อร่วมกัน ซึ่งก็มีทั้งแบบต่อขาแคโทดร่วมกันเรียกว่า แบบแคโทดร่วม (Common cathode) และแบบต่อขาแอโนดร่วมกันเรียกว่า แบบแอโนดร่วม (Common anode) การขับให้แอลอีดีเจ็ดส่วนแบบแคโทดร่วมสว่างจะต้องจ่ายไฟลบเข้าที่ขาร่วม แล้วจ่ายไฟบวกเข้าที่ขาแอโนดซึ่งก็คือขาของแต่ละส่วน ดังแสดงในรูปที่ 2.7 ในขณะที่แอลอีดีเจ็ดส่วนแบบแอดโนดร่วมจะต้องจ่ายไฟบวกเข้าที่ขาร่วม แล้วจ่ายไฟลบเข้าที่ขาแคโทดซึ่งเป็นขาของแต่ละส่วนดังแสดงในรูปที่ 2.8

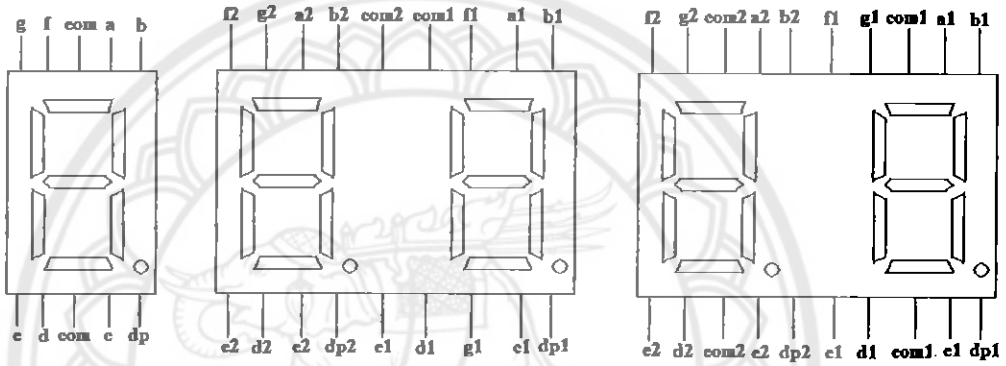



**รูปที่ 2.7 การต่อแอลอีดีเจ็ดส่วนแบบแอโนดร่วม**




**รูปที่ 2.8** การต่อแอลอีดีเจ็ดส่วนแบบแคโทดร่วม

แอลอีดีเจ็ดส่วนมีจำหน่ายทั้งแบบตัวเดี่ยว ตัวคู่ และแบบที่มากกว่า 2 หลัก แต่ที่นิยมใช้งานและหาได้ง่ายมี 2 แบบคือ แบบตัวเดี่ยวและแบบตัวคู่ โดยมีการจัดขาดังรูปที่ 2.9




**รูปที่ 2.9** แอลอีดีเจ็ดส่วนแบบตัวเดี่ยวและแบบตัวคู่

จะเห็นได้ว่าแอลอีดีเจ็ดส่วนแบบตัวเดี่ยวมีขาต่อใช้งาน 10 ขาคือ ขา a, b, c, d, e, f, g, dp และขาร่วม (Common) ซึ่งมี 2 ขา ถ้าเป็นแอลอีดีเจ็ดส่วนแบบตัวคู่จะมีขาต่อใช้งาน 20 ขา แบ่งเป็นขา a, b, c, d, e, f, g และ dp อย่างละ 2 ขารวม 16 ขาและขาร่วมอีกหลักละ 2 ขา การต่อขาร่วมของแต่ละหลักทั้ง 2 ขานั้น สามารถต่อใช้งานเพียงขาเดียวได้ เนื่องจากในโครงสร้างภายในแอลอีดีเจ็ดส่วนขาร่วมนี้ต่อถึงกันอยู่แล้ว

ในการขับแอลอีดีเจ็ดส่วนแบบหลักเดี่ยว ไมโครคอนโทรลเลอร์ MCS-51 ไม่ควรนำมาขับแอลอีดีเจ็ดส่วนโดยตรง เพราะความสามารถในการจ่ายกระแสเอาท์พุตรวมไม่สูงมากนัก จึงต้องอาศัยไอซีบัฟเฟอร์มาช่วยในการขับแอลอีดี อาทิ ไอซีเบอร์ 74HC541 และที่เอาท์พุตของไอซีบัฟเฟอร์ที่ต่อกับแอลอีดีเจ็ดส่วนต้องมีตัวต้านทานจำกัดกระแสให้แก่แอลอีดีในทุกส่วน

# บทที่ 3
# การออกแบบและการทำงานเครื่องควบคุมอุณหภูมิของน้ำ

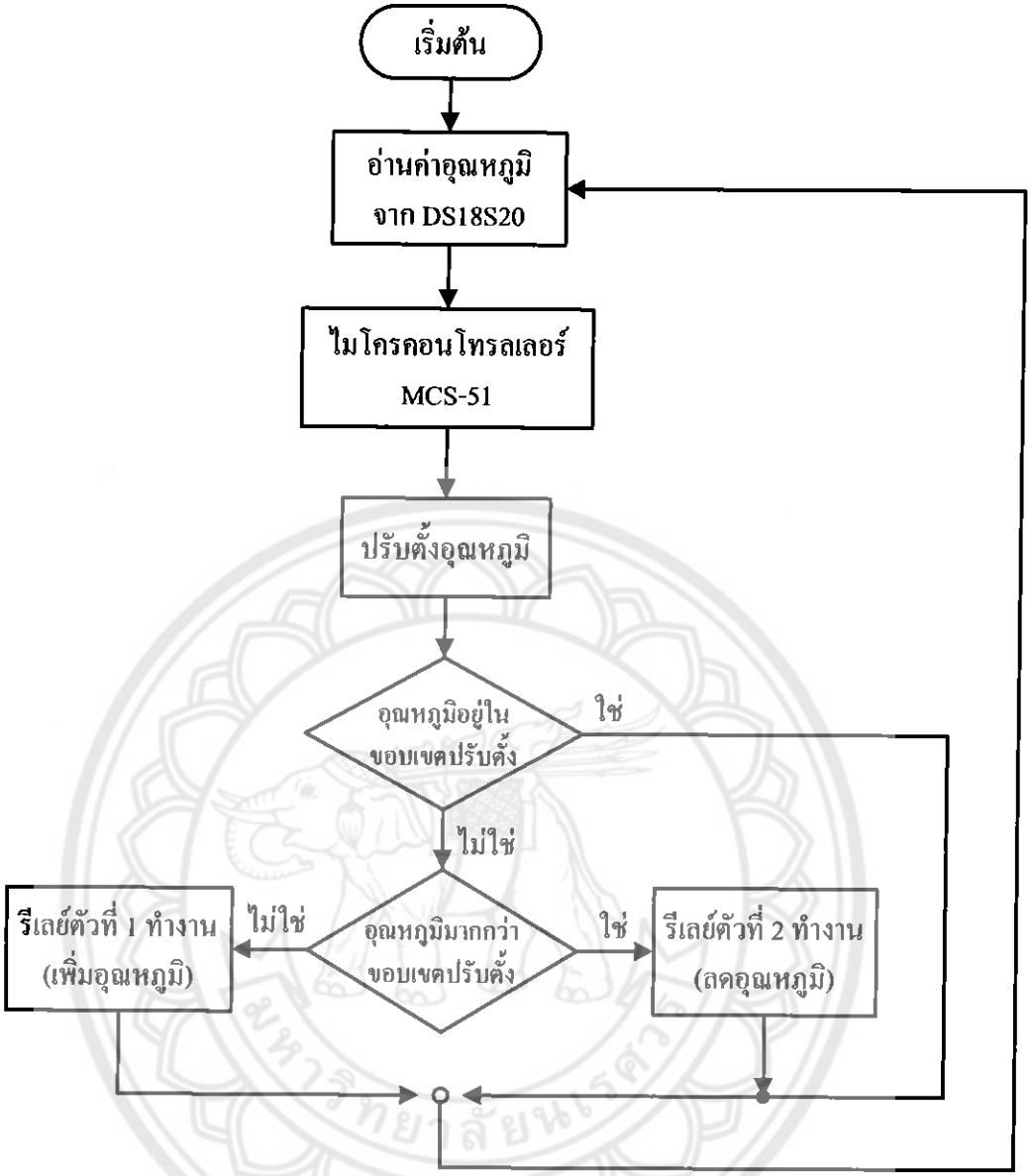
ในบทนี้ได้นำเครื่องควบคุมอุณหภูมิของน้ำมาอธิบายทั้งในส่วนประกอบและวิธีใช้ การทำงานของโปรแกรมเพื่อควบคุมการทำงาน และการทำงานของวงจรภายในเครื่อง แบ่งได้ 4 วงจร คือ วงจรไมโครคอนโทรลเลอร์ MCS-51 วงจรแอลอีดีเจ็ดส่วน วงจรควบคุมการทำงานอุปกรณ์ปรับอุณหภูมิ และวงจรไอซี DS18S20

## 3.1 ส่วนประกอบและวิธีการใช้เครื่องควบคุมอุณหภูมิของน้ำ

ในหัวข้อนี้เป็นการแนะนำเครื่องควบคุมอุณหภูมิของน้ำให้เข้าใจถึงโครงสร้างภายนอกและภายในตัวเครื่อง พร้อมทั้งได้อธิบายวิธีใช้งานของเครื่อง

### 3.1.1 ส่วนประกอบเครื่องควบคุมอุณหภูมิของน้ำ

การทำงานเครื่องควบคุมอุณหภูมิของน้ำจะมีการทำงานเป็นลำดับขั้นตอน เมื่อเริ่มการทำงานโดยมีการอ่านค่าอุณหภูมิของน้ำได้จากไอซี DS18S20 แล้วจะส่งสัญญาณไปยังวงจรไมโครคอนโทรลเลอร์ MCS-51 เพื่อประมวลผลและทำการปรับอุณหภูมิในช่วงที่ต้องการ เมื่ออุณหภูมิจากไอซี DS18S20 มากกว่าค่าที่ปรับตั้ง ในกรณีที่อุณหภูมิน้อยกว่าค่าที่ปรับตั้งจะส่งสัญญาณไปส่งให้รีเลย์ตัวที่ 1 ทำงานถ้าอุณหภูมิมากกว่าค่าที่ปรับตั้งจะส่งไปสัญญาณยังรีเลย์ตัวที่ 2 ให้ทำงานและเมื่ออุณหภูมิอยู่ในช่วงที่ปรับตั้งไว้รีเลย์จะไม่มีการทำงาน แล้วจะส่งสัญญาณกลับไปยังไอซี DS18S20 เพื่ออ่านค่าอุณหภูมิต่อไป แสดงดังรูปที่ 3.1




**รูปที่ 3.1** การทำงานเครื่องควบคุมอุณหภูมิของน้ำ

เครื่องควบคุมอุณหภูมิของน้ำมีส่วนประกอบภายนอกดังรูปที่ 3.2

รูปที่ 3.2 ส่วนประกอบภายนอกของเครื่องควบคุมอุณหภูมิของน้ำ

โดยที่

"1" คือ หน้าจอแสดงค่าอุณหภูมิ

"2" คือ สวิตช์ตั้งค่า

"3" คือ สัญญาณไฟแอลอีดีแสดงการทำงานของอุปกรณ์ปรับอุณหภูมิ

"4" คือ สวิตช์เปิดการทำงานอุปกรณ์ลดอุณหภูมิโดยตรง

"5" คือ สวิตช์เปิดการทำงานอุปกรณ์เพิ่มอุณหภูมิโดยตรง

"6" คือ เต้าเสียบอุปกรณ์ปรับลดอุณหภูมิ

"7" คือ เต้าเสียบอุปกรณ์ปรับเพิ่มอุณหภูมิ

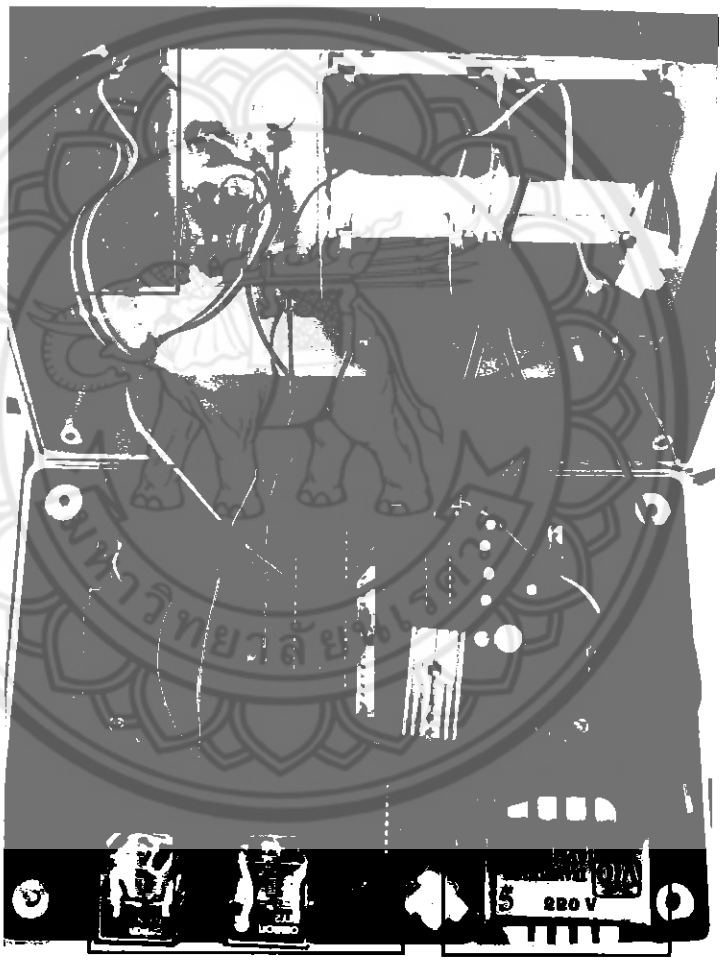
"8" คือ ฟิวส์ป้องกันกระแส

"9" คือ เต้าเสียบ

"10" คือ ตัวอ่านอุณหภูมิ

"11" คือ สัญญาณไฟแอลอีดีเตือนการตั้งค่าอุณหภูมิ

ส่วนประกอบภายในของเครื่องควบคุมอุณหภูมิ แสดงในรูปที่ 3.3



**รูปที่ 3.3** ส่วนประกอบภายในของเครื่องควบคุมอุณหภูมิของน้ำ

โดยที่

"1" คือ วงจรแอลอีดีเจ็ดส่วน

"2" คือ วงจรไอซี DS18S20

"3" คือ วงจรไมโครคอนโทรลเลอร์

"4" คือ วงจรควบคุมการทำงานอุปกรณ์ปรับอุณหภูมิ

"5" คือ หม้อแปลงแรงดัน

### 3.1.2 วิธีการใช้เครื่องควบคุมอุณหภูมิของน้ำ

วิธีการใช้เครื่องควบคุมอุณหภูมิของน้ำมีขั้นตอนดังนี้

1) ติดตั้งเครื่องควบคุมอุณหภูมิของน้ำไว้บริเวณตู้ปลา เสียบปลั๊กเพื่อเริ่มการทำงาน สังเกตที่หน้าจอแสดงอุณหภูมิและสัญญาณไฟแอลอีดีจะมีการทำงาน
2) นำตัวตรวจจับอุณหภูมิจุ่มน้ำในตู้ปลา รอ 1 นาทีเพื่อให้ตัวอ่านอุณหภูมิปรับอุณหภูมิให้คงที่
3) ติดตั้งอุปกรณ์ปรับอุณหภูมิทั้ง 2 ชนิดกับเครื่องควบคุมอุณหภูมิของน้ำ
4) ตั้งค่าอุณหภูมิโดยกดสวิตช์ SET สังเกตที่หน้าจอแสดงค่าอุณหภูมิจะเปลี่ยนเป็นโหมดตั้งค่าอุณหภูมิ สังเกตที่หน้าจอแสดงค่าอุณหภูมิจะกระพริบ สามารถกดสวิตช์ UP เพื่อเพิ่มค่าอุณหภูมิครั้งละ 0.5 หรือกดสวิตช์ DOWN เพื่อลดค่าอุณหภูมิครั้งละ 0.5
5) เมื่อตั้งค่าอุณหภูมิเสร็จแล้วให้กดสวิตช์ SET อีกครั้งเพื่อเข้าโหมดปรับตั้งค่าอุณหภูมิที่ต้องการรักษา ในโหมดนี้จะค่าการปรับตั้งให้เลือก 3 ค่า คือ 1.0, 1.5 และ 2.0 สามารถกดสวิตช์ UP เพื่อเลือกปรับตั้งค่าอุณหภูมิที่ต้องการรักษา ซึ่งค่าปรับตั้งอุณหภูมิที่ต้องการรักษาจะเป็นตัวกำหนดขอบเขตอุณหภูมิที่ต้องการควบคุม ตัวอย่างเช่น ตั้งค่าอุณหภูมิไว้ที่ 25.0 องศาเซลเซียส ปรับตั้งค่าอุณหภูมิที่ต้องการรักษาไว้ที่ 1.0 เครื่องควบคุมอุณหภูมิของน้ำจะสั่งให้อุปกรณ์เพิ่มอุณหภูมิทำงานที่ 24.0 องศาเซลเซียส และจะสั่งให้อุปกรณ์เพิ่มอุณหภูมิทำงานที่ 26.0 องศาเซลเซียส เป็นต้น เมื่อปรับตั้งค่าอุณหภูมิที่ต้องการควบคุมเสร็จแล้วให้กดสวิตช์ SET เพื่อออกสู่โหมดทำงานปกติเช่นเดิม
6) สามารถสั่งให้ให้อุปกรณ์ปรับอุณหภูมิทำงานได้โดยตรงจากการกดสวิตช์ที่ช่องทำงานของอุปกรณ์ปรับอุณหภูมินั้น

หมายเหตุ: หากยังไม่มีการตั้งค่าอุณหภูมิหรือเกิดการรีเซตของเครื่องควบคุมอุณหภูมิของน้ำที่สัญญาณไฟแอลอีดีเตือนการตั้งค่าจะสว่างขึ้นเพื่อเตือนให้ผู้ใช้งานทำการตั้งค่า และเมื่อกดสวิตช์ SET สัญญาณไฟแอลอีดีเตือนการตั้งค่าจะดับลงทันที

เมื่อรู้ถึงส่วนประกอบและเข้าใจวิธีการใช้งานเครื่องควบคุมอุณหภูมิของน้ำแล้ว ในหัวข้อต่อไปได้อธิบายการทำงานของโปรแกรมเพื่อให้เข้าใจถึงขั้นตอนการทำงานของโปรแกรมเพื่อควบคุมการทำงานของเครื่อง

## 3.2 การทำงานของโปรแกรมเพื่อควบคุมการทำงานของเครื่องควบคุมอุณหภูมิ

ในการเขียนโปรแกรมเพื่อควบคุมการทำงานเครื่องควบคุมอุณหภูมิของน้ำสามารถแบ่งออกเป็น 3 ขั้นตอน โดยการทำงานเริ่มจากขั้นตอนการอ่านค่าอุณหภูมิจากไอซี DS18S20 ซึ่งในขั้นตอนนี้มีการติดต่อกับไอซี DS18S20 เพื่อนำค่าอุณหภูมิที่ได้ไปแสดงที่แอลอีดีเจ็ดส่วน ขั้นตอนต่อมาเป็นขั้นตอนการตั้งค่าอุณหภูมิ ในขั้นตอนนี้เป็นการให้ผู้ใช้งานกำหนดค่าอุณหภูมิที่ต้องการรักษา และในขั้นตอนการควบคุมการทำงานรีเลย์จะทำการปรับอุณหภูมิให้อยู่ในระดับที่ต้องการรักษา ซึ่งขั้นตอนการทำงานของเครื่องควบคุมอุณหภูมิของน้ำแสดงได้ดังรูปที่ 3.4

**รูปที่ 3.4** แผนภาพการทำงานของเครื่องควบคุมอุณหภูมิ

### 3.2.1 ขั้นตอนการอ่านค่าอุณหภูมิจากไอซี DS18S20

ข้อมูลอุณหภูมิที่ได้จากไอซี DS18S20 เป็นเลขฐาน 2 จำนวน 2 ไบต์ (1 ไบต์เท่ากับ 8 บิต) ไบต์ที่ 1 จะแสดงค่าเหมือนกันทุกบิต โดยถ้าแสดงเป็น "0" หมายความว่าอุณหภูมิที่อ่านได้มีค่าเป็นบวก แต่ถ้าแสดงเป็น "1" หมายความว่าอุณหภูมิที่อ่านได้เป็นลบ ต่อมาในไบต์ที่ 2 จะบอกค่าอุณหภูมิจำนวนสองหลักในรูปของเลขฐานสิบ และบิตสุดท้ายคือ บิต 0 บอกค่าอุณหภูมิของหลักทศนิยม โดยถ้าแสดงเป็น "0" ค่าอุณหภูมิในหลักทศนิยมจะมีค่าเป็น 0 หากแสดงค่าเป็น "1" ค่าอุณหภูมิในหลักทศนิยมจะเป็น 5 ดังแสดงในตารางที่ 3.1 และเมื่อแปลงข้อมูลอุณหภูมิจากไอซี DS18S20 เป็นค่าอุณหภูมิสองหลักแล้วจะนำค่าที่ได้ไปแสดงที่แอลอีดีเจ็ดส่วน

**ตารางที่ 3.1** ข้อมูล 2 บิต ที่ได้จากไอซี DS18S20

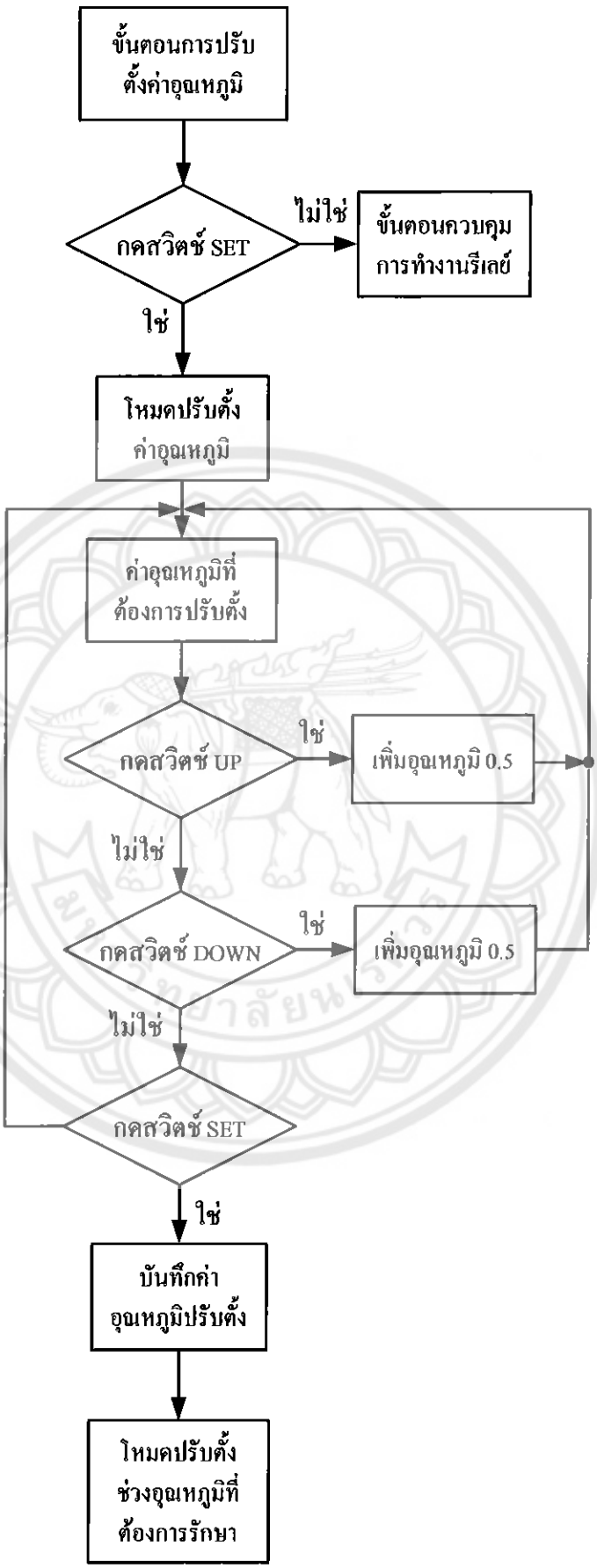
| ไบต์ที่ 2 | บิต 7 | บิต 6 | บิต 5 | บิต 4 | บิต 3 | บิต 2 | บิต 1 | บิต 0 |
|---|---|---|---|---|---|---|---|---|
| | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ | $2^{-1}$ |
| ไบต์ที่ 1 | บิต 15 | บิต 14 | บิต 13 | บิต 12 | บิต 11 | บิต 10 | บิต 9 | บิต 8 |
| | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

### 3.2.2 ขั้นตอนการปรับตั้งค่าอุณหภูมิ

เมื่อจบขั้นตอนการอ่านค่าอุณหภูมิจากไอซี DS18S20 แล้ว ในขั้นตอนต่อมาคือการตั้งค่าปรับตั้งค่าอุณหภูมิ ในขั้นตอนนี้เริ่มจากการตรวจสอบสัญญาณอินพุตที่พอร์ต P.3.0 ซึ่งเป็นสัญญาณจากการกดสวิตช์ SET ถ้าหากไม่มีการกดสวิตช์ SET โปรแกรมจะข้ามไปที่ขั้นตอนต่อไปคือขั้นตอนการควบคุมการทำงานรีเลย์ แต่หากมีการกดสวิตช์ SET โปรแกรมจะเข้าสู่การปรับตั้งค่าอุณหภูมิ ซึ่งในการปรับตั้งนี้โปรแกรมสามารถแบ่งการตั้งค่าออกเป็น 2 โหมด คือ

1) โหมดปรับตั้งค่าอุณหภูมิ

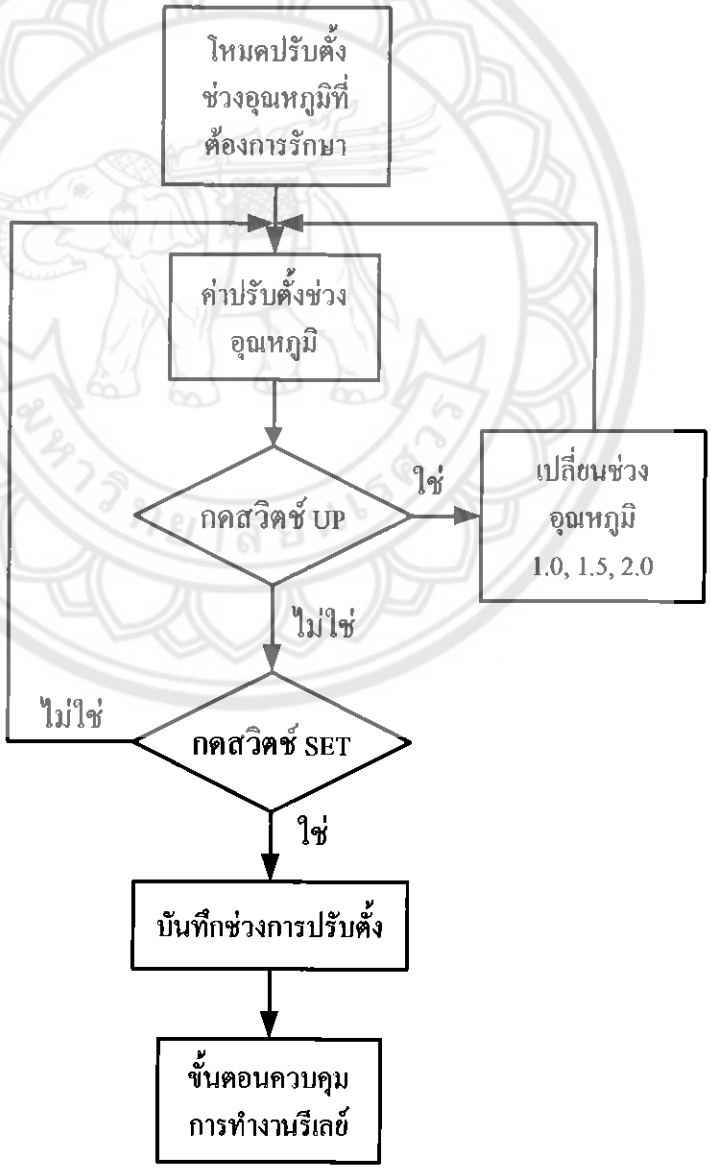
การกำหนดค่าอุณหภูมิที่ต้องการเป็นโหมดแรกของขั้นตอนการปรับตั้ง การทำงานเริ่มจากการนำค่าอุณหภูมิที่ปรับตั้งไว้มาแสดงที่หน้าจอโดยหากเป็นการตั้งค่าครั้งแรกโปรแกรมจะตั้งค่าไว้ที่ 25.0 องศาเซลเซียส จากนั้นโปรแกรมจะมีการตรวจสอบสัญญาณอินพุตทั้งหมด 3 อินพุต คือ สัญญาณที่พอร์ต P.3.1 เป็นสัญญาณจากสวิตช์ UP เมื่อมีการกดสวิตช์ค่าอุณหภูมิจะเพิ่มขึ้น 0.5 องศาเซลเซียส สัญญาณที่พอร์ต P.3.2 เป็นสัญญาณจากสวิตช์ DOWN เมื่อมีการกดสวิตช์ค่าอุณหภูมิจะลดขึ้น 0.5 องศาเซลเซียส และสัญญาณที่พอร์ต P.3.0 เป็นสัญญาณจากสวิตช์ SET เมื่อมีการกดสวิตช์โปรแกรมจะเก็บค่าอุณหภูมิที่ปรับตั้งแล้วเปลี่ยนไปยังโหมดตั้งปรับตั้งช่วงอุณหภูมิที่ต้องการรักษา ซึ่งการทำงานของโหมดนี้สามารถแสดงเป็นแผนภาพได้ดังรูปที่ 3.5




**รูปที่ 3.5** แผนภาพการทำงานโหมดการปรับตั้งค่าอุณหภูมิ

**2) โหมดปรับตั้งช่วงอุณหภูมิที่ต้องการรักษา**

โหมดต่อมาเป็นการตั้งขอบเขตเพื่อควบคุมช่วงอุณหภูมิที่ต้องการรักษา โดยการทำงานของโปรแกรมจะเริ่มจากการนำค่าปรับตั้งช่วงอุณหภูมิที่ต้องการรักษามาแสดงที่หน้าจอแสดงค่าอุณหภูมิ จากนั้นโปรแกรมจะตรวจสอบสัญญาณอินพุต 2 สัญญาณ คือ สัญญาณที่พอร์ต P.3.1 ซึ่งเป็นสัญญาณจากสวิตช์ UP เมื่อมีการกดสวิตช์ค่าปรับตั้งช่วงอุณหภูมิจะเปลี่ยนค่า โดยการปรับตั้งช่วงอุณหภูมิมีค่าเป็น 1.0, 1.5 และ 2.0 ตามลำดับ และสัญญาณที่พอร์ต P.3.0 เป็นสัญญาณจากสวิตช์ SET เมื่อมีการกดสวิตช์โปรแกรมจะเก็บค่าปรับตั้งช่วงอุณหภูมิแล้วจบการทำงานในขั้นตอนการปรับตั้งค่าอุณหภูมิ จากนั้นจะเริ่มการทำงานในขั้นตอนต่อไปคือขั้นตอนการควบคุมการทำงานรีเลย์ซึ่งการทำงานของโหมดนี้สามารถแสดงเป็นแผนภาพได้ดังรูปที่ 3.6




**รูปที่ 3.6** แผนภาพการทำงานโหมดปรับตั้งช่วงอุณหภูมิที่ต้องการรักษา

### 3.2.3 ขั้นตอนควบคุมการทำงานรีเลย์

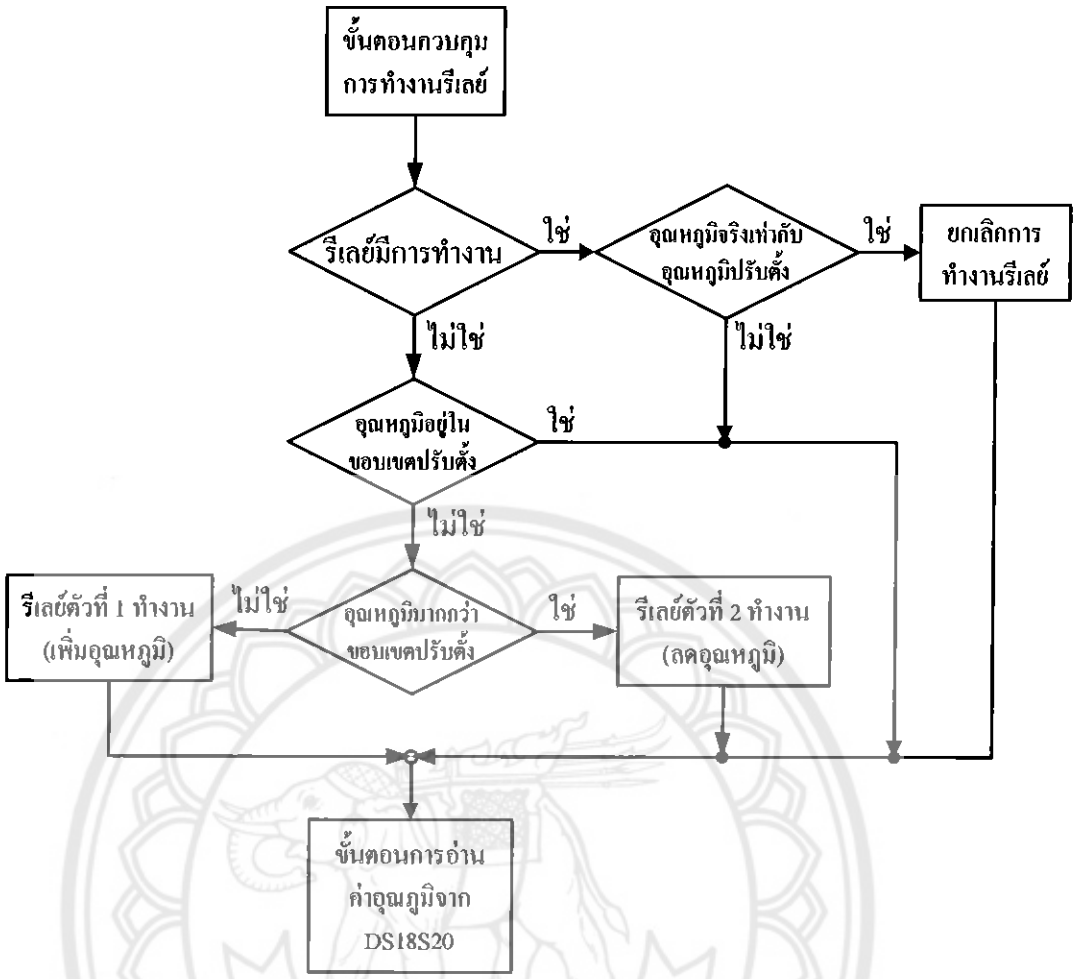
เมื่อมีการตั้งปรับค่าอุณหภูมิแล้วโปรแกรมจะตรวจสอบก่อนว่ามีการส่งสัญญาณไปที่พอร์ต P.2.1 หรือ P.2.2 อยู่หรือไม่ ถ้าหากมีการส่งสัญญาณอยู่แสดงว่ามีการทำงานของรีเลย์อยู่ โปรแกรมจะทำการเปรียบเทียบค่าอุณหภูมิที่ได้จากไอซี DS18S20 กับค่าอุณหภูมิที่ปรับตั้งค่าไว้ โดยผลจากการเปรียบเทียบแบ่งได้ 2 กรณีคือ

1) ค่าอุณหภูมิจากไอซี DS18S20 กับค่าอุณหภูมิที่ตั้งค่าไว้มีค่าไม่เท่ากัน โปแกรมจะจบการทำงานในขั้นตอนการควบคุมการทำงานรีเลย์
2) ค่าอุณหภูมิจากไอซี DS18S20 กับค่าอุณหภูมิที่ตั้งค่าไว้มีค่าเท่ากัน โปรแกรมจะหยุดการส่งสัญญาณให้ทั้งพอร์ต P.2.1 และ P.2.2 แล้วจบการทำงานในขั้นตอนการควบคุมการทำงานรีเลย์

แต่หากการตรวจสอบของโปรแกรมไม่พบว่ามีการส่งสัญญาณไปที่พอร์ต P.2.1 หรือ P.2.2 โปรแกรมจะนำค่าอุณหภูมิที่ได้จากไอซี DS18S20 มาเปรียบเทียบกับขอบเขตการปรับตั้งอุณหภูมิ โดยผลจากการเปรียบเทียบแบ่งได้ 3 กรณีคือ

1) ค่าอุณหภูมิจากไอซี DS18S20 มีค่าน้อยกว่าขอบเขตการปรับตั้งอุณหภูมิที่ต้องการรักษา โปรแกรมจะส่งสัญญาณไปที่พอร์ต P.2.1 เพื่อให้รีเลย์ตัวที่ 1 (เพิ่มอุณหภูมิ) ทำงาน
2) ค่าอุณหภูมิจากไอซี DS18S20 มีค่ามากกว่าขอบเขตการปรับตั้งอุณหภูมิที่ต้องการรักษา โปรแกรมจะส่งสัญญาณไปที่พอร์ต P.2.2 เพื่อให้รีเลย์ตัวที่ 2 (ลดอุณหภูมิ) ทำงาน
3) ค่าอุณหภูมิจากไอซี DS18S20 มีค่าอยู่ในช่วงขอบเขตการปรับตั้งอุณหภูมิที่ต้องการรักษาโปรแกรมจะไม่มีการส่งสัญญาณให้กับพอร์ต P.2.1 และ P.2.2

เมื่อโปรแกรมจบการเปรียบเทียบใน 2 กรณีแรกหรือใน 3 กรณีหลังเรียบร้อยแล้ว การทำงานจะจบขั้นตอนควบคุมการทำงานรีเลย์แล้วกลับไปเริ่มต้นที่ขั้นตอนแรกคือขั้นตอนการอ่านค่าอุณหภูมิจากไอซี DS18S20 ขั้นตอนการควบคุมรีเลย์แสดงเป็นแผนภาพได้ดังรูปที่ 3.7




**รูปที่ 3.7** แผนภาพการทำงานขั้นตอนการควบคุมการทำงานรีเลย์

วงจรต่าง ๆ ภายในเครื่องควบคุมอุณหภูมิของน้ำถูกออกแบบให้ทำงานสัมพันธ์กัน โดยในหัวข้อต่อไป จะอธิบายหลักการออกแบบและการทำงานของวงจรต่าง ๆ ที่อยู่ในเครื่องควบคุมอุณหภูมิของน้ำ

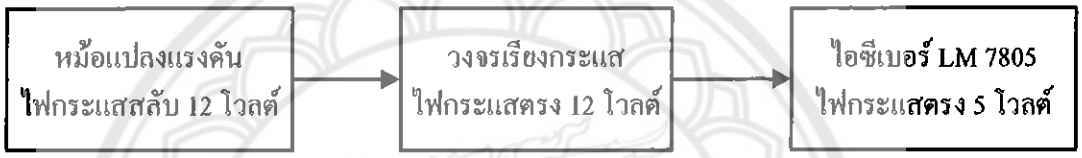
## 3.3 วงจรไมโครคอนโทรลเลอร์ MCS-51

วงจรไมโครคอนโทรลเลอร์ MCS-51 เป็นส่วนหลักในการควบคุมการทำงานของเครื่องควบคุมอุณหภูมิของน้ำ โดยในวงจรนี้สามารถแบ่งออกเป็น 3 ส่วน ส่วนแรกคือ ส่วนวงจรควบคุมแรงดัน เป็นส่วนรับไฟกระแสสลับ 12 โวลต์ จากหม้อแปลงแรงดันเข้ามาแล้วเปลี่ยนระดับแรงดันเป็นไฟกระแสตรง 5 โวลต์ เพื่อนำไปใช้กับวงจรไมโครคอนโทรลเลอร์ MCS-51 ส่วนที่สองคือส่วนวงจรป้อนข้อมูลเป็นวงจรแปลงระดับสัญญาณจาก RS-232 เป็นระดับสัญญาณ TTL แล้ว

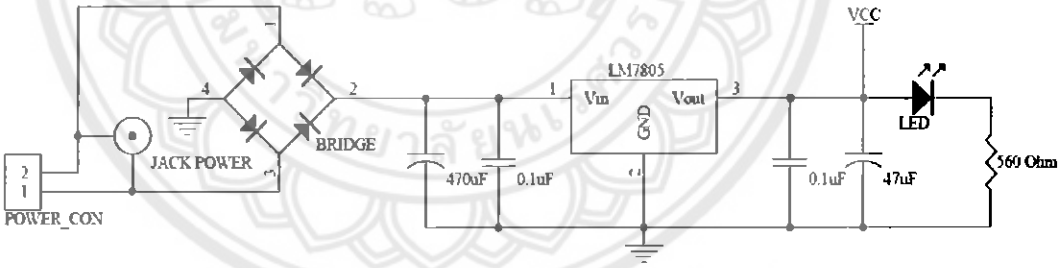
ป้อนเข้าที่วงจรไมโครคอนโทรลเลอร์ และส่วนที่สามคือส่วนวงจรไมโครคอนโทรลเลอร์ เป็นวงจรหลักในการควบคุมการทำงานของเครื่องควบคุมอุณหภูมิของน้ำ

### 3.3.1 ส่วนวงจรควบคุมแรงดัน

ในส่วนนี้การทำงานเริ่มต้นจากรับแรงดันจากหม้อแปลงไฟกระแสสลับ 12 โวลต์ จากแหล่งจ่ายไฟมาผ่านวงจรเรียงกระแสเพื่อเปลี่ยนจากไฟกระแสสลับเป็นไฟกระแสตรง แล้วลดระดับแรงดันลงเหลือ 5 โวลต์ โดยผ่านไอซี LM7805 เมื่อได้ไฟกระแสตรง 5 โวลต์ แล้ว ถูกนำไปใช้กับวงจรไมโครคอนโทรลเลอร์ต่อไป โดยการทำงานสามารถเขียนเป็นแผนภาพได้ดังรูปที่ 3.8 และวงจรควบคุมแรงดันแสดงดังรูปที่ 3.9




รูปที่ 3.8 แผนภาพกรอบการทำงานในส่วนวงจรควบคุมแรงดัน




รูปที่ 3.9 วงจรควบคุมแรงดัน

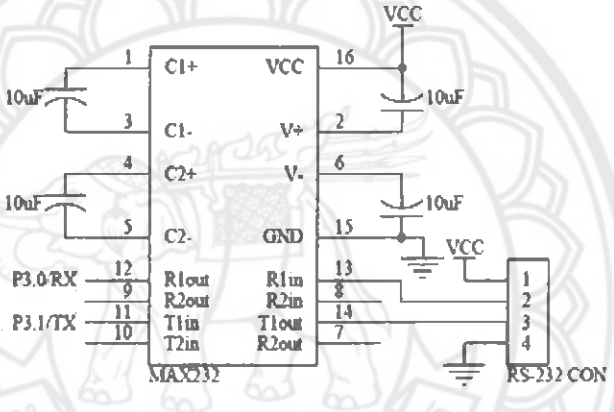
### 3.3.2 ส่วนวงจรป้อนข้อมูล

เมื่อเขียนโปรแกรมเพื่อควบคุมการทำงานด้วยภาษาแอสเซมบลีในคอมพิวเตอร์แล้ว ขั้นต่อมาคือการนำข้อมูลนั้นส่งไปที่ไมโครคอนโทรลเลอร์ โดยข้อมูลที่ส่งมาจากคอมพิวเตอร์จะมีการเรียงระดับสัญญาณแบบ RS-232 ผ่านไอซี MAX232 เพื่อเรียงสัญญาณใหม่ให้เป็นระดับสัญญาณ TTL แล้วส่งข้อมูลเข้าไปยังไมโครคอนโทรลเลอร์ การ

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร


ทำงานในส่วนของวงจรป้อนข้อมูลนี้เขียนเป็นแผนภาพได้ดังรูปที่ 3.10 และวงจรป้อนข้อมูลแสดงดังรูปที่ 3.11

**รูปที่ 3.10** แผนภาพกรอบการทำงานในส่วนวงจรป้อนข้อมูล

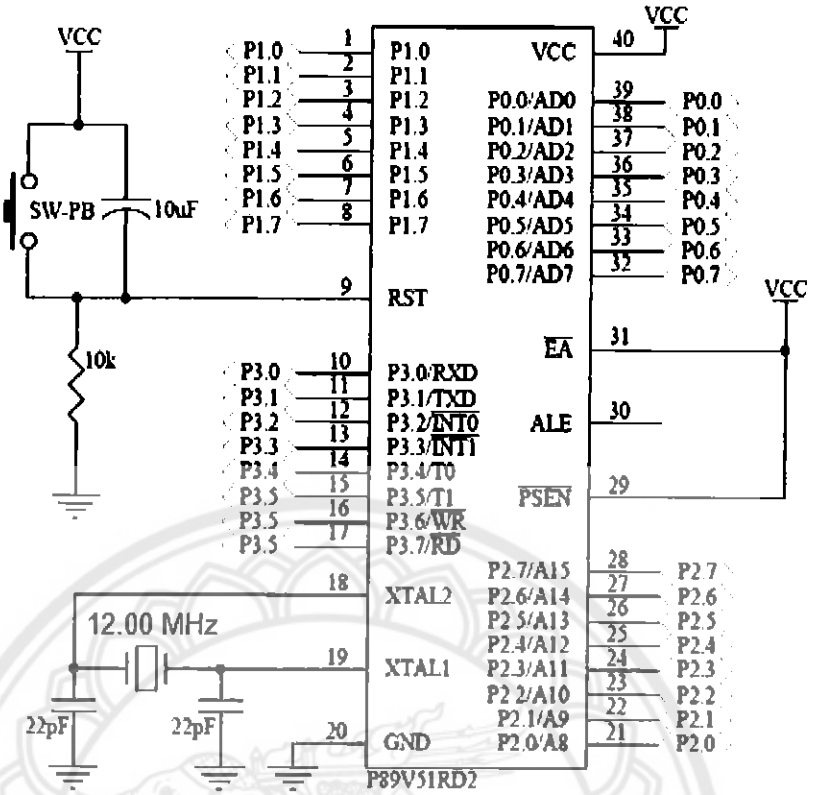



**รูปที่ 3.11** วงจรป้อนข้อมูล

### 3.3.3 ส่วนวงจรไมโครคอนโทรลเลอร์

ในส่วนนี้เป็นวงจรหลักในการประมวลผลและควบคุมการทำงานของเครื่องควบคุมอุณหภูมิของน้ำ ซึ่งการควบคุมจะเป็นในรูปแบบการรับส่งสัญญาณผ่านพอร์ต P.0–P.3 โดยวงจรไมโครคอนโทรลเลอร์แสดงดังรูปที่ 3.12

1574764
ม.ร.
ณ3420
2553




รูปที่ 3.12 วงจรไมโครคอนโทรลเลอร์ [5]

มีการกำหนดการทำงานที่พอร์ตอินพุตและเอาท์พุตของไมโครคอนโทรลเลอร์ แสดงได้ดังตารางที่ 3.2 และตารางที่ 3.3 ตามลำดับ

ตารางที่ 3.2 พอร์ตอินพุตของไมโครคอนโทรลเลอร์

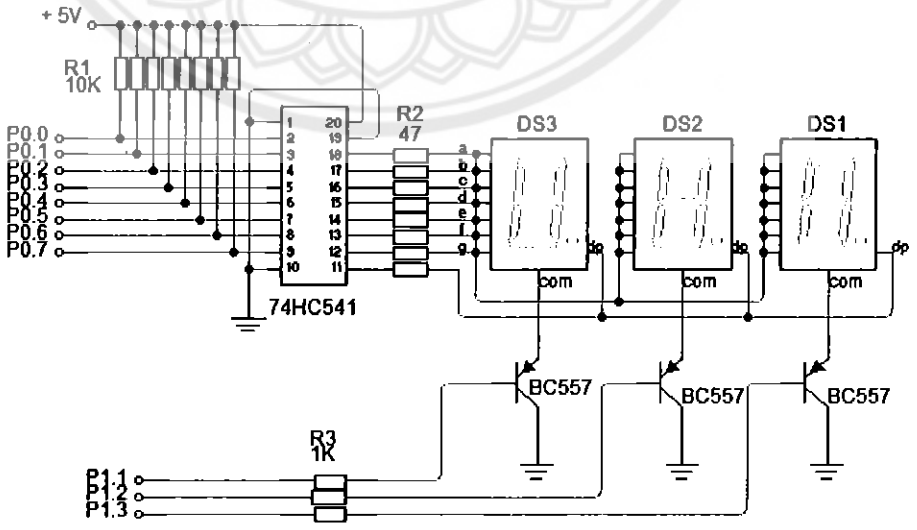
| พอร์ต | การทำงาน |
|---|---|
| P.2.7 | รับค่าอุณหภูมิจากไอซี DS18S20 |
| P.3.0 | รับสัญญาณจากโหมด SET |
| P.3.1 | รับสัญญาณจากโหมด UP |
| P.3.2 | รับสัญญาณจากโหมด DOWN |

**ตารางที่ 3.3** พอร์ตเอาท์พุตของไมโครคอนโทรลเลอร์

| พอร์ต | การทำงาน |
|---|---|
| P.0 | ส่งสัญญาณให้แอลอีดีเจ็ดส่วน |
| P.1.1 | ส่งสัญญาณให้แอลอีดีเจ็ดส่วนหลักสิบ |
| P.1.2 | ส่งสัญญาณให้แอลอีดีเจ็ดส่วนหลักหน่วย |
| P.1.3 | ส่งสัญญาณให้แอลอีดีเจ็ดส่วนหลักทศนิยม |
| P.2.1 | ส่งสัญญาณให้รีเลย์ตัวที่ 1 (เพิ่มอุณหภูมิ) |
| P.2.2 | ส่งสัญญาณให้รีเลย์ตัวที่ 2 (ลดอุณหภูมิ) |

## 3.4 วงจรแอลอีดีเจ็ดส่วน

วงจรแอลอีดีเจ็ดส่วนในเครื่องควบคุมอุณหภูมิของน้ำมีการต่อกับไมโครคอนโทรลเลอร์เพื่อแสดงข้อมูลตัวเลขที่พอร์ต P0 โดยผ่านไอซี 74HC541 ซึ่งเป็นไอซีรักษาระดับแรงดันและช่วยป้องกันไมโครคอนโทรลเลอร์จากกระแสลัดวงจรหากมีการลัดวงจรที่แอลอีดีเจ็ดส่วน และมีการกำหนดการแสดงผลในแต่ละหลักของแอลอีดีเจ็ดส่วนที่พอร์ต P.1.1-P.1.3 วงจรแอลอีดีเจ็ดส่วนที่สร้างขึ้นแสดงดังรูปที่ 3.13



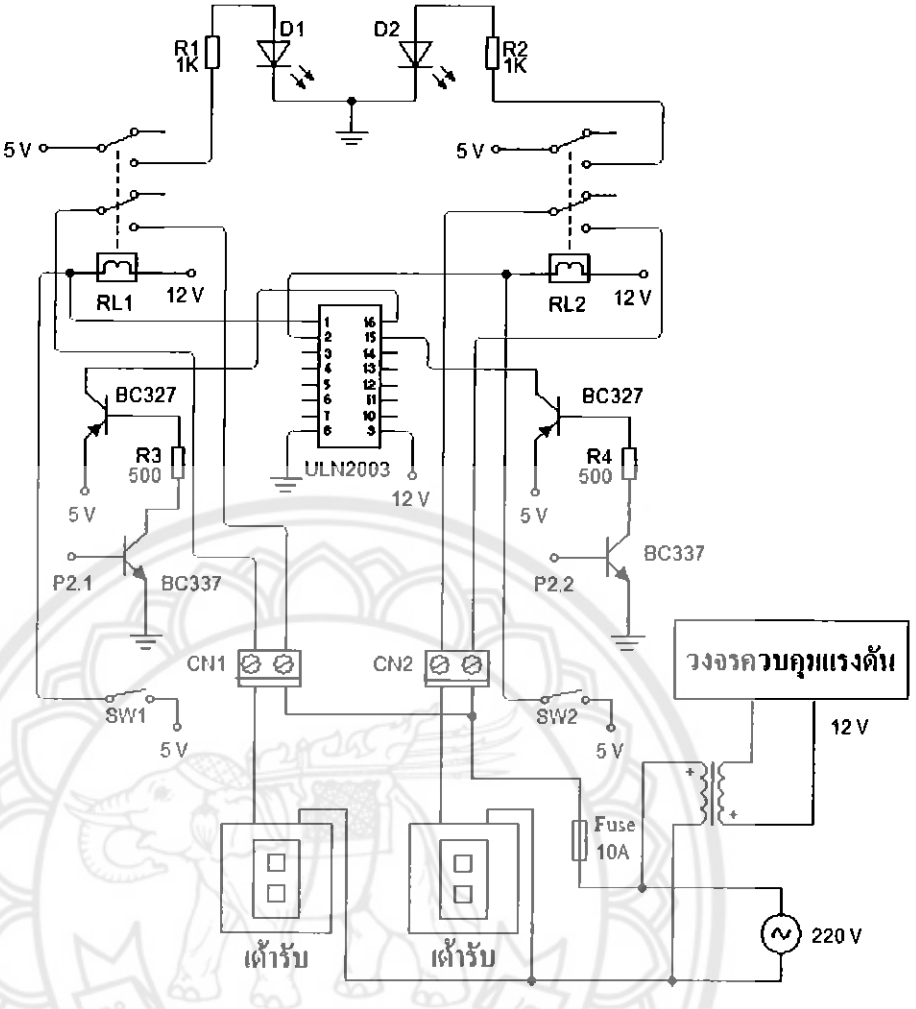

**รูปที่ 3.13** วงจรแอลอีดีเจ็ดส่วน

## 3.5 วงจรควบคุมการทำงานอุปกรณ์ปรับอุณหภูมิ

เมื่อไมโครคอนโทรลเลอร์ MCS-51 ส่งสัญญาณเอาท์พุตมายังพอร์ต P.2.1หรือ P.2.2 ที่ต่ออยู่กับขาเบสของทรานซิสเตอร์ BC337 ทำให้มีแรงดัน 5 โวลต์ มาตกคร่อมที่ขาเบสกับขาอีมิตเตอร์ ส่งผลให้ทรานซิสเตอร์นำกระแสจากคอลเลคเตอร์ไปอีมิตเตอร์ ซึ่งขาคอลเลคเตอร์ของ BC337 ต่ออยู่กับขาเบสของ BC327 ส่งผลให้มีแรงดัน -5 โวลต์ ตกคร่อมที่ขาเบสและขาอิมิตเตอร์ของ BC327 ทำให้ขาคอลเลคเตอร์ของ BC327 ซึ่งต่ออยู่กับไอซี ULN2003 ได้รับแรงดัน 5 โวลต์ ซึ่งไอซี ULN2003 มีหน้าที่เปลี่ยนระดับแรงดันจาก 0 โวลต์ เป็น 12 โวลต์ในสภาวะปกติ และเปลี่ยนระดับแรงดันจาก 5 โวลต์ เป็น 0 โวลต์ เพื่อทำให้ขดลวดของรีเลย์เกิดความต่างศักย์จนเกิดแรงเหนี่ยวนำทำให้หน้าสัมผัสของรีเลย์ทั้ง 2 หน้าสัมผัสปกติเปิดเชื่อมต่อวงจรทำให้เกิดการทำงานดังนี้

1) สัญญาณไฟแอลอีดีติดสว่างเพื่อแจ้งการทำงานอุปกรณ์ปรับอุณหภูมิสว่าง
2) ทำให้มีไฟจากแหล่งจ่ายไปยังโหลดหรืออุปกรณ์ปรับอุณหภูมิโดยผ่านเต้ารับ

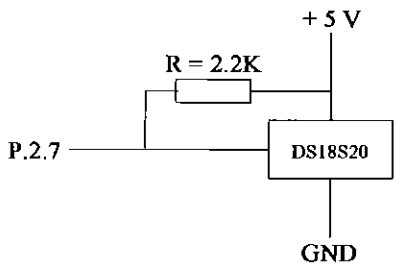
นอกจากนี้เรายังสามารถควบคุมให้รีเลย์ทำงานโดยกดสวิตช์เพื่อสั่งการทำงานได้โดยตรง ซึ่งวงจรควบคุมการทำงานอุปกรณ์ปรับอุณหภูมิแสดงดังรูปที่ 3.14




**รูปที่ 3.14** วงจรควบคุมการทำงานอุปกรณ์ปรับอุณหภูมิ

## 3.6 วงจรไอซี DS18S20

ไอซี DS18S20 ที่ขา VDD ต่อกับไฟกระแสตรง 5 โวลต์ ขา GND ต่อกับสายดิน และขา DQ ต่อกับพอร์ต P.2.7 ของไมโครคอนโทรลเลอร์เพื่อส่งค่าอุณหภูมิที่วัดได้ไปให้ไมโครคอนโทรลเลอร์ วงจรไอซี DS18S20 แสดงดังรูปที่ 3.15




**รูปที่ 3.15** วงจรDS18S20

# บทที่ 4
# ผลการทดสอบ

หลังจากที่ได้สร้างเครื่องควบคุมอุณหภูมิของน้ำแล้ว ผู้ดำเนินโครงงานได้ทำการทดสอบการทำงานของเครื่องเพื่อวัดประสิทธิภาพการทำงานของเครื่องในการอ่านค่าอุณหภูมิ และความสามารถในรักษาระดับอุณหภูมิ โดยการทดสอบแบ่งออกเป็น 2 การทดสอบ ดังนี้

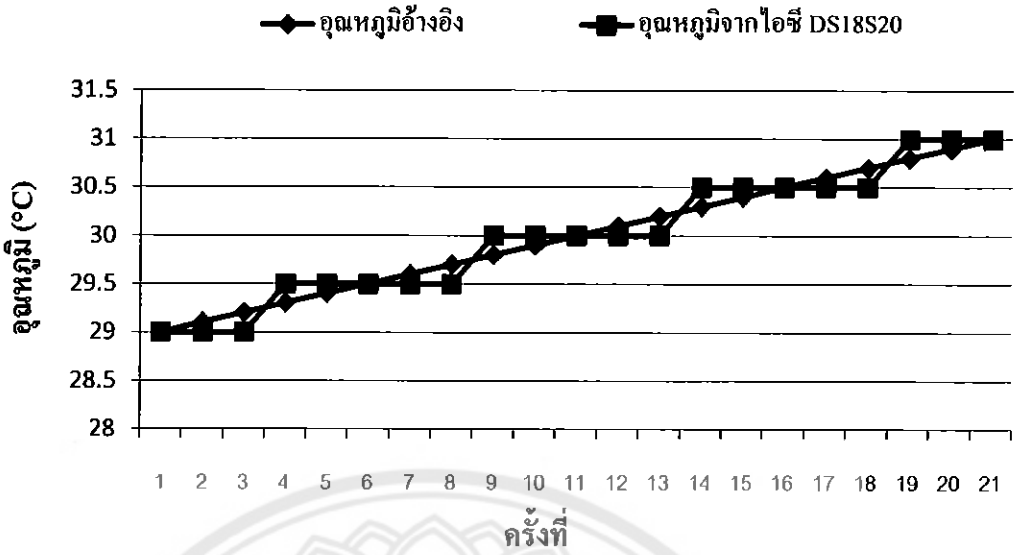
1) ทดสอบการอ่านค่าอุณหภูมิของไอซี DS18S20
2) ทดสอบความสามารถในการปรับระดับอุณหภูมิ

ผู้ดำเนินโครงงานได้เลือกอุปกรณ์ที่นำมาใช้ทดสอบกับเครื่องควบคุมอุณหภูมิของน้ำทั้ง 3 การทดสอบดังนี้

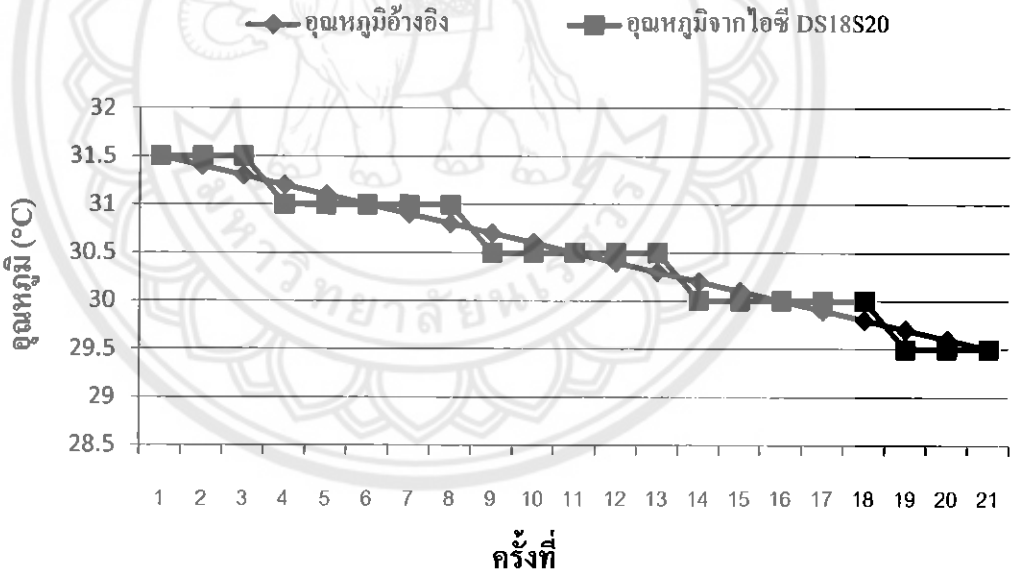
1) ตู้ปลาขนาด 20 นิ้ว (20x10x11 นิ้ว) เนื่องจากเป็นขนาดที่พอเหมาะสำหรับการเลี้ยงปลาสวยงามขนาดเล็กและขนาดกลาง ซึ่งในการทดลองมีปริมาณน้ำในตู้ 36 ลิตร
2) อุปกรณ์ปรับเพิ่มอุณหภูมิที่ใช้คือเครื่องทำความร้อน ที่มีขนาดที่เหมาะสมกับตู้ขนาด 20–24 นิ้ว
3) อุปกรณ์ปรับลดอุณหภูมิที่ใช้คือพัดลมระบายความร้อน ที่มีขนาดที่เหมาะสมกับตู้ขนาด 20–24 นิ้ว

## 4.1 การทดสอบการอ่านค่าอุณหภูมิของไอซี DS18S20

การใช้งานโดยทั่วไปของไอซี DS18S20 คือวัดอุณหภูมิของอากาศ แต่ในโครงงานนี้ได้นำไอซีดังกล่าวมาประยุกต์การใช้งานกับการวัดค่าอุณหภูมิของน้ำจึงต้องมีวัสดุครอบตัวไอซีไว้ เพื่อไม่ให้ไอซีเสียหายจากการจุ่มน้ำ แต่อาจส่งผลให้การอ่านค่าอุณหภูมิมีความคลาดเคลื่อนเนื่องจากตัวไอซีไม่ได้อ่านค่าอุณหภูมิของน้ำโดยตรง ดังนั้นผู้ดำเนินโครงงานจึงทำการทดสอบการอ่านค่าอุณหภูมิของตัวไอซีขึ้น ซึ่งในการทดสอบได้เลือกทดสอบที่ช่วงอุณหภูมิเพิ่มขึ้นจากอุณหภูมิเริ่มต้นของน้ำและที่ช่วงอุณหภูมิลดลงจากอุณหภูมิเริ่มต้นของน้ำ โดยเลือกใช้อุปกรณ์วัดอุณหภูมิของน้ำ Ocean free prise digital thermometer ที่ได้รับการรับรองมาตรฐานจาก ISO 9001:2000 มาเป็นตัวเปรียบเทียบความถูกต้องของอุณหภูมิ ผลการทดสอบการอ่านอุณหภูมิที่ช่วงอุณหภูมิเพิ่มขึ้นจากอุณหภูมิเริ่มต้นของน้ำแสดงดังรูปที่ 4.1 และผลการทดสอบการอ่านอุณหภูมิที่ช่วงอุณหภูมิลดลงจากอุณหภูมิเริ่มต้นของน้ำแสดงดังรูปที่ 4.2




**รูปที่ 4.1** ผลการทดลองการอ่านค่าอุณหภูมิของไอซี DS18S20 ที่ช่วงอุณหภูมิเพิ่มขึ้น




**รูปที่ 4.2** ผลการทดลองการอ่านค่าอุณหภูมิของไอซี DS18S20 ที่ช่วงอุณหภูมิลดลง

จากผลทดสอบดังรูปที่ 4.1 และรูปที่ 4.2 โดยเปรียบเทียบอุณหภูมิอ้างอิงโดยใช้ Ocean free prise กับอุณหภูมิจากไอซี DS18S20 ที่สร้างขึ้น มีค่าสัมประสิทธิ์สหสัมพันธ์เท่ากับ 0.9765 ซึ่งได้จากสมการที่ (1) บ่งบอกว่าข้อมูลทั้งสองมีค่าใกล้เคียงกันนั่นคือตัวตรวจจับอุณหภูมิที่สร้างขึ้นมีความแม่นยำและเชื่อถือได้ เพราะหากพิจารณาถึงผลกระทบที่เกิดขึ้นจากการอ่านค่าอุณหภูมิของเครื่องนั้นส่งผลต่ออุณหภูมิในตู้ปลาน้อยมาก

$$r_{x,y} = \frac{n\sum xy - \sum x \sum y}{\sqrt{\{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]\}}} \qquad (1)$$

โดยที่

$r_{x,y}$ คือ ค่าสัมประสิทธิ์สหสัมพันธ์

x คือ อุณหภูมิอ้างอิง

y คือ อุณหภูมิจากไอซี DS18S20

n คือ จำนวนครั้ง

## 4.2 การทดสอบความสามารถในการปรับอุณหภูมิ

เนื่องจากปลาสวยงามโดยทั่วไปมักชอบอุณหภูมิระหว่าง 25–32 องศาเซลเซียส และปลาแต่ละชนิดมีความทนต่อการเปลี่ยนแปลงอุณหภูมิที่แตกต่างกันต่างกัน ซึ่งการเปลี่ยนแปลงอุณหภูมิเพียง 2–5 องศาเซลเซียส ก็เริ่มส่งผลกระทบต่อปลาสวยงาม ดังนั้นในการทดสอบนี้ผู้ดำเนินโครงงานได้ทดสอบประสิทธิภาพการรักษาระดับอุณหภูมิของเครื่องควบคุมอุณหภูมิของน้ำ โดยในการทดสอบผู้ดำเนินโครงงานได้กำหนดให้ทดสอบการรักษาระดับอุณหภูมิที่มากกว่าอุณหภูมิเริ่มต้นของน้ำในตู้ 2 องศา และน้อยกว่าอุณหภูมิเริ่มต้นของน้ำในตู้ 2 องศา แล้วบันทึกผลการทดลองดังตารางที่ 4.1 และตารางที่ 4.2

ตารางที่ 4.1 ผลการทดสอบรักษาระดับอุณหภูมิที่ต่ำกว่าอุณหภูมิน้ำ 2 องศา

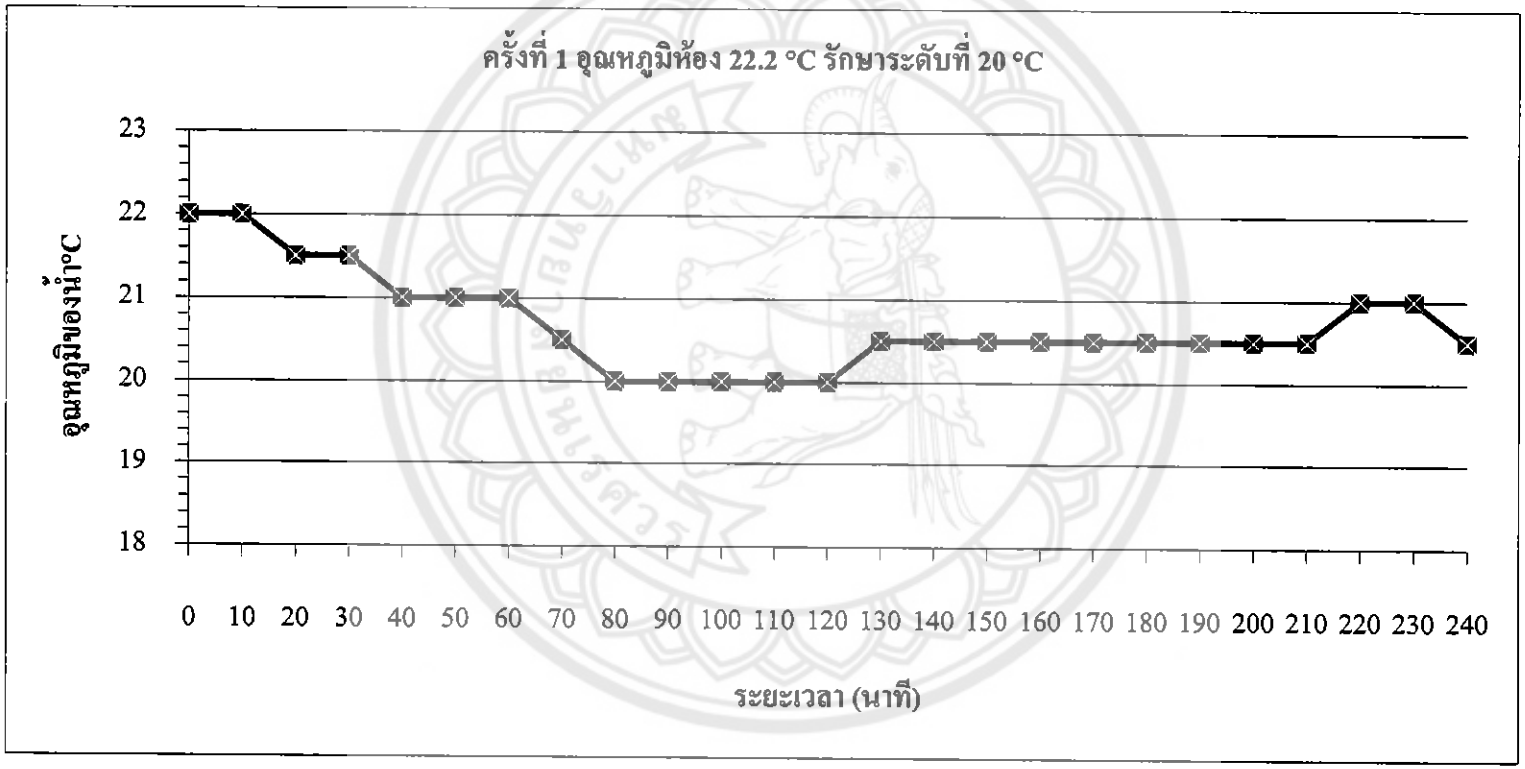
| เวลา (นาที) | อุณหภูมิ (°C) ของห้อง | อุณหภูมิ (°C) ของน้ำ | ระดับอุณหภูมิ(°C) ที่ต้องการ | สภาวะการทำงาน |
|---|---|---|---|---|
| 0 | 22.2 | 22 | 20 | ทำความเย็น |
| 10 | 22.2 | 22 | 20 | ทำความเย็น |
| 20 | 22.2 | 21.5 | 20 | ทำความเย็น |
| 30 | 22.2 | 21.5 | 20 | ทำความเย็น |
| 40 | 22.2 | 21 | 20 | ทำความเย็น |
| 50 | 22.2 | 21 | 20 | ทำความเย็น |
| 60 | 22.2 | 21 | 20 | ทำความเย็น |
| 70 | 22.2 | 20.5 | 20 | ทำความเย็น |
| 80 | 22.2 | 20 | 20 | ไม่มีการทำงาน |
| 90 | 22.2 | 20 | 20 | ไม่มีการทำงาน |
| 100 | 22.2 | 20 | 20 | ไม่มีการทำงาน |
| 110 | 22.1 | 20 | 20 | ไม่มีการทำงาน |
| 120 | 22.1 | 20 | 20 | ไม่มีการทำงาน |
| 130 | 22.1 | 20.5 | 20 | ไม่มีการทำงาน |
| 140 | 22.1 | 20.5 | 20 | ไม่มีการทำงาน |
| 150 | 22 | 20.5 | 20 | ไม่มีการทำงาน |
| 160 | 22 | 20.5 | 20 | ไม่มีการทำงาน |
| 170 | 22 | 20.5 | 20 | ไม่มีการทำงาน |
| 180 | 22 | 20.5 | 20 | ไม่มีการทำงาน |
| 190 | 22 | 20.5 | 20 | ไม่มีการทำงาน |
| 200 | 22 | 20.5 | 20 | ไม่มีการทำงาน |
| 210 | 22 | 20.5 | 20 | ไม่มีการทำงาน |
| 220 | 22 | 21 | 20 | ทำความเย็น |
| 230 | 22 | 21 | 20 | ทำความเย็น |
| 240 | 22 | 20.5 | 20 | ทำความเย็น |

**ตารางที่ 4.2** ผลการทดสอบรักษาระดับอุณหภูมิที่มากกว่าอุณหภูมิน้ำ 2 องศา

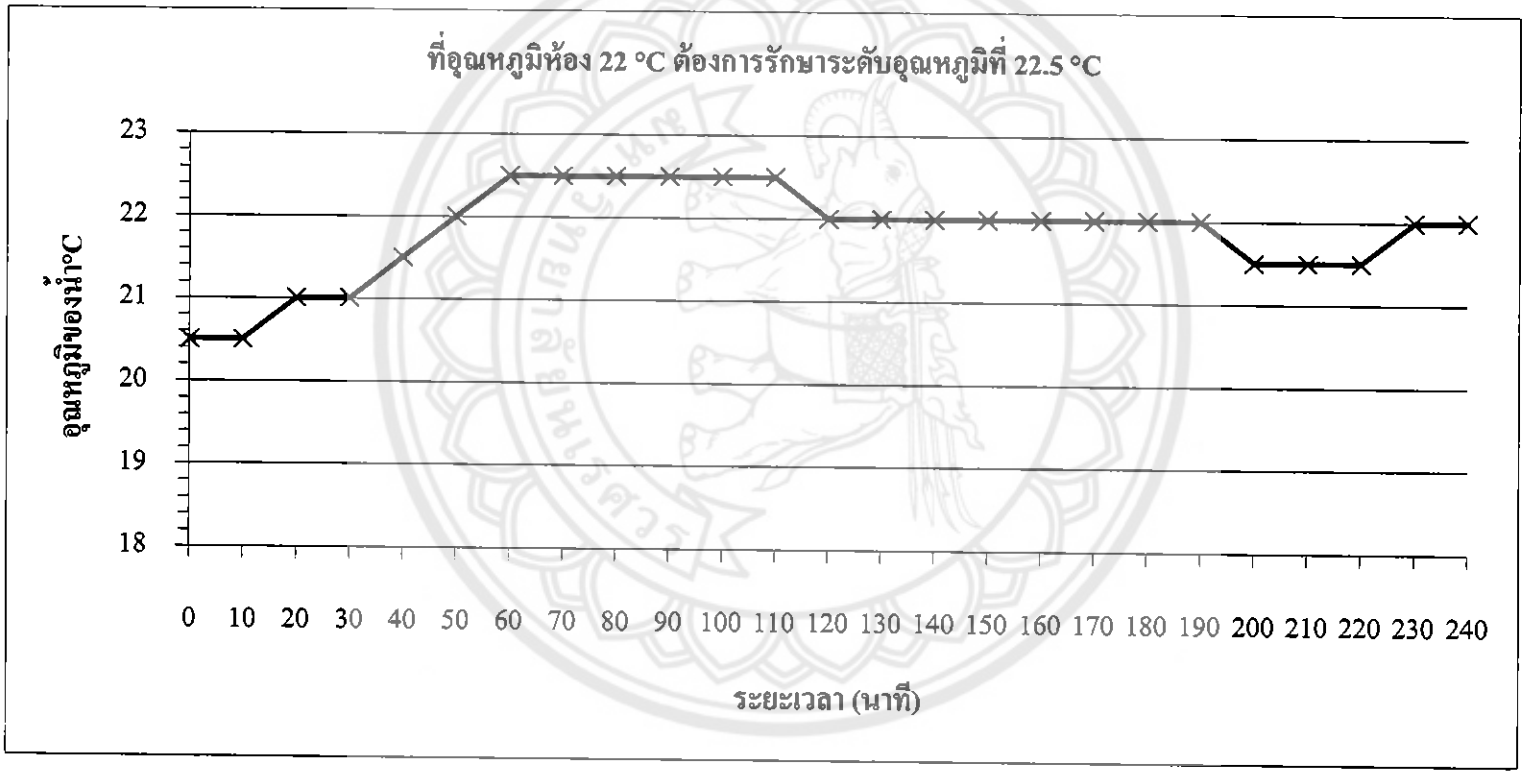
| เวลา (นาที) | อุณหภูมิ (°C) ของห้อง | อุณหภูมิ (°C) ของน้ำ | ระดับอุณหภูมิ(°C) ที่ต้องการ | สภาวะการทำงาน |
|---|---|---|---|---|
| 0 | 22 | 20.5 | 22.5 | ทำความร้อน |
| 10 | 22 | 20.5 | 22.5 | ทำความร้อน |
| 20 | 22 | 21 | 22.5 | ทำความร้อน |
| 30 | 22 | 21 | 22.5 | ทำความร้อน |
| 40 | 22 | 21.5 | 22.5 | ทำความร้อน |
| 50 | 22 | 22 | 22.5 | ทำความร้อน |
| 60 | 22 | 22.5 | 22.5 | ไม่มีการทำงาน |
| 70 | 22 | 22.5 | 22.5 | ไม่มีการทำงาน |
| 80 | 21.9 | 22.5 | 22.5 | ไม่มีการทำงาน |
| 90 | 21.9 | 22.5 | 22.5 | ไม่มีการทำงาน |
| 100 | 21.9 | 22.5 | 22.5 | ไม่มีการทำงาน |
| 110 | 21.9 | 22.5 | 22.5 | ไม่มีการทำงาน |
| 120 | 21.9 | 22 | 22.5 | ไม่มีการทำงาน |
| 130 | 21.8 | 22 | 22.5 | ไม่มีการทำงาน |
| 140 | 21.8 | 22 | 22.5 | ไม่มีการทำงาน |
| 150 | 21.8 | 22 | 22.5 | ไม่มีการทำงาน |
| 160 | 21.8 | 22 | 22.5 | ไม่มีการทำงาน |
| 170 | 21.8 | 22 | 22.5 | ไม่มีการทำงาน |
| 180 | 21.8 | 22 | 22.5 | ไม่มีการทำงาน |
| 190 | 21.8 | 22 | 22.5 | ไม่มีการทำงาน |
| 200 | 21.8 | 21.5 | 22.5 | ทำความร้อน |
| 210 | 21.8 | 21.5 | 22.5 | ทำความร้อน |
| 220 | 21.8 | 21.5 | 22.5 | ทำความร้อน |
| 230 | 21.8 | 22 | 22.5 | ทำความร้อน |
| 240 | 21.8 | 22 | 22.5 | ทำความร้อน |

จากการทดสอบการปรับลดอุณหภูมิลงด้วยการใช้พัดลมระบายอากาศเป่าที่ผิวน้ำได้ผลการทดสอบดังตารางที่ 4.1 โดยเมื่อเริ่มทดสอบที่อุณหภูมิห้อง 22.2 องศาเซลเซียส อุณหภูมิของน้ำ 22 องศาเซลเซียส มีการเริ่มทำงานของพัดลมระบายอากาศ และเมื่อระยะเวลา 80 นาที ระดับอุณหภูมิลดลงถึงค่าที่ต้องการรักษาคือที่อุณหภูมิของน้ำ 20 องศาเซลเซียส พัดลมจึงหยุดทำงาน และในระยะเวลา 220 นาที อุณหภูมิของน้ำเพิ่มขึ้นเป็น 21 องศาเซลเซียส พัดลมระบายอากาศจึงเริ่มทำงานอีกครั้งเนื่องจากอุณหภูมิเพิ่มขึ้นถึงขอบเขตที่ต้องการรักษาระดับอุณหภูมิ ซึ่งการเปลี่ยนแปลงอุณหภูมิของน้ำสามารถแสดงความสัมพันธ์ระหว่างระดับอุณหภูมิที่ลดลงกับเวลาได้ดังรูปที่ 4.3

จากการทดสอบการปรับเพิ่มอุณหภูมิด้วยการใช้เครื่องทำความร้อนได้ผลการทดสอบดังตารางที่ 4.2 โดยเมื่อเริ่มทดสอบที่อุณหภูมิห้อง 22 องศาเซลเซียส อุณหภูมิของน้ำ 20.5 องศาเซลเซียส มีการเริ่มทำงานของเครื่องทำความร้อน และเมื่อและเมื่อระยะเวลา 60 นาที ระดับอุณหภูมิเพิ่มขึ้นถึงค่าที่ต้องการรักษาที่อุณหภูมิของน้ำ 22.5 องศาเซลเซียส เครื่องทำความร้อนจึงหยุดทำงาน และในระยะเวลา 200 นาที อุณหภูมิของน้ำลดลงเป็น 21.5 องศาเซลเซียส เครื่องทำความร้อนจึงเริ่มทำงานอีกครั้งเนื่องจากอุณหภูมิลดลงถึงขอบเขตที่ต้องการรักษาระดับอุณหภูมิ ซึ่งการเปลี่ยนแปลงอุณหภูมิของน้ำสามารถแสดงความสัมพันธ์กราฟระหว่างระดับอุณหภูมิที่เพิ่มขึ้นกับเวลาได้ดังรูปที่ 4.4




**รูปที่ 4.3** กราฟความสัมพันธ์ระหว่างระดับอุณหภูมิที่ลดลงกับเวลา




**รูปที่ 4.4** กราฟความสัมพันธ์ระหว่างระดับอุณหภูมิที่เพิ่มขึ้นกับเวลา

จากกราฟในรูปที่ 4.3 แสดงให้เห็นว่าการลดลงของอุณหภูมิจนอยู่ในระดับที่ปรับตั้งไว้อยู่ที่ระยะเวลา 80 นาที และจากกราฟในรูปที่ 4.4 แสดงให้เห็นว่าการเพิ่มขึ้นของอุณหภูมิจนอยู่ในระดับที่ปรับตั้งไว้อยู่ที่ระยะเวลา 60 นาที เมื่อพิจารณาการเปลี่ยนแปลงของกราฟทั้ง 2 กราฟปรากฏว่าการเปลี่ยนแปลงในช่วงลดอุณหภูมิมีอัตราการเปลี่ยนแปลงช้ากว่าอัตราการเปลี่ยนแปลงในช่วงเพิ่มอุณหภูมิ สาเหตุที่ทำให้การเปลี่ยนแปลงในช่วงลดอุณหภูมิช้ากว่าเนื่องจากอุปกรณ์ที่นำมาใช้เพื่อปรับลดอุณหภูมิคือ พัดลมระบายอากาศที่มีหลักการปรับลดอุณหภูมิโดยดูดอากาศรอบข้างมาเป่าลงที่ผิวน้ำ ดังนั้นประสิทธิภาพการปรับลดอุณหภูมิด้วยพัดลมระบายอากาศจึงขึ้นอยู่กับอุณหภูมิแวดล้อมเป็นหลัก ในขณะที่การปรับเพิ่มอุณหภูมิมีการเปลี่ยนแปลงเร็วกว่าเนื่องจากอุปกรณ์ที่นำมาใช้ปรับเพิ่มอุณหภูมิคือ เครื่องทำความร้อนที่มีหลักการปรับเพิ่มอุณหภูมิโดยการส่งกระแสไฟฟ้าผ่านที่ขดลวดจนเกิดความร้อนแผ่อุณหภูมิให้น้ำได้โดยตรง ดังนั้นประสิทธิภาพการปรับเพิ่มอุณหภูมิของเครื่องทำความร้อนจึงขึ้นอยู่กับการทำความร้อนของเครื่องทำความร้อนโดยตรง

จากการทดสอบเครื่องควบคุมอุณหภูมิทั้ง 2 การทดสอบแสดงให้เห็นว่าเครื่องควบคุมอุณหภูมิของน้ำสามารถนำไปใช้รักษาระดับอุณหภูมิของน้ำให้อยู่ในช่วงที่ปลาสวยงามต้องการได้

# บทที่ 5
# สรุปผลและข้อเสนอแนะ

จากการดำเนินโครงงานสามารถสรุปผล ชี้แจงปัญหาที่เกิดขึ้นในระหว่างการดำเนินงาน รวมทั้งเสนอแนวทางแก้ปัญหา พร้อมให้ข้อเสนอแนะในการนำโครงงานไปพัฒนาต่อไป

## 5.1 สรุปผลการดำเนินโครงงาน

ในโครงงานนี้ได้ศึกษาเกี่ยวกับการเปลี่ยนแปลงอุณหภูมิของน้ำกับปลาสวยงามเพื่อหาระดับอุณหภูมิที่เหมาะสมกับปลาสวยงามทั่วไป จากการศึกษาพบว่าปลาสวยงามมีผลกระทบอย่างมากหากอาศัยอยู่ในน้ำที่มีอุณหภูมิไม่เหมาะสม ผู้ดำเนินโครงงานจึงสร้างเครื่องควบคุมอุณหภูมิของน้ำ โดยใช้ไมโครคอนโทรลเลอร์มาทำหน้าที่ประมวลผลจากการเปรียบเทียบค่าอุณหภูมิจากไอซี DS18S20 กับค่าที่กำหนดโดยผู้ใช้งาน แล้วสั่งให้พัดลมระบายอากาศหรือเครื่องทำความร้อนทำงานเพื่อปรับอุณหภูมิของน้ำให้อยู่ในช่วงที่กำหนด

เครื่องควบคุมอุณหภูมิของน้ำที่ผู้ดำเนินโครงงานสร้างขึ้นได้ผ่านการทดสอบความสามารถของตัวไอซี DS18S20 ที่ถูกนำมาใช้อ่านค่าอุณหภูมิ สรุปได้ว่ามีถูกต้องของการอ่านค่าอุณหภูมิอยู่ในช่วงที่ยอมรับได้ และทดสอบการปรับระดับอุณหภูมิ สรุปได้ว่าการปรับเพิ่มอุณหภูมิด้วยเครื่องทำความร้อนสามารถปรับอุณหภูมิได้รวดเร็ว เนื่องจากการปรับอุณหภูมิของเครื่องทำความร้อนขึ้นอยู่กับการทำความร้อนของเครื่องทำความร้อนโดยตรง แต่การปรับลดอุณหภูมิด้วยพัดลมระบายอากาศสามารถปรับอุณหภูมิได้ช้ากว่า เนื่องจากการปรับอุณหภูมิด้วยพัดลมระบายอากาศขึ้นอยู่กับอุณหภูมิรอบข้าง จากผลการทดสอบทั้ง 2 การทดสอบสรุปได้ว่าเครื่องควบคุมอุณหภูมิของน้ำสามารถปรับระดับอุณหภูมิของน้ำให้อยู่ในช่วงอุณหภูมิที่เหมาะสมกับปลาสวยงามได้

## 5.2 ปัญหาและแนวทางแก้ไข

เครื่องควบคุมอุณหภูมิของน้ำที่สร้างขึ้นสามารถอ่านอุณหภูมิได้ละเอียดสูงสุดเพียง 0.5 องศาเซลเซียสเท่านั้น ซึ่งเป็นข้อจำกัดของความสามารถในการอ่านค่าอุณหภูมิของตัวไอซี DS18S20 จึงทำให้ความละเอียดในการรักษาระดับอุณหภูมิสามารถควบคุมได้ละเอียดเพียง

0.5 องศาเซลเซียสเช่นกัน แนวทางแก้ไขคือสามารถเปลี่ยนตัวอ่านค่าอุณหภูมิให้มีความสามารถในการอ่านค่าอุณหภูมิที่มีความละเอียดมากขึ้น

## 5.3 แนวทางในการพัฒนาต่อไป

นอกจากการปัจจัยอุณหภูมิที่มีผลต่อสุขภาพของปลา และยังมีอีกหลายปัจจัยที่มีส่งผลต่อปลาที่อาจทำให้ปลาเป็นโรค รูปร่างและสีสันไม่สวยงาม และไม่กินอาหาร โดยปัจจัยที่ทำให้เกิดปัญหาเหล่านี้ คือ น้ำไม่สะอาด ซึ่งปัจจัยนี้สามารถควบคุมได้เช่นกัน ดังนั้นแนวทางในการพัฒนาต่อไปควรพัฒนาเครื่องควบคุมอุณหภูมิของน้ำให้สามารถควบคุมความสะอาดของน้ำโดยเพิ่มเครื่องวัดความขุ่นของน้ำมาตรวจสอบความขุ่นเพื่อเปลี่ยนถ่ายน้ำให้สะอาดยิ่งขึ้น

# เอกสารอ้างอิง

[1] อุณหภูมิประเทศไทย สืบค้นเมื่อ 1 มีนาคม 2554, จาก http://thai.tourismthail.org/about-thailand/weather/

[2] เชือน กัลมาพิจิตร์. ตำราการเลี้ยงปลาตู้. โอเดียนสโตร์. กรุงเทพฯ. 2545

[3] อุดม จีนประดับ. ไมโครคอนโทรลเลอร์ MCS-51. สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ. กรุงเทพฯ. 2541

[4] อดุลย์ กัลยาแก้ว. อุปกรณ์อิเล็กทรอนิกส์และวงจร. ห้างหุ้นส่วนจำกัด วี.เจ. พริ้นติ้ง. 2546

[5] วรพจน์ กรแก้ววัฒนกุล และชัยวัฒน์ ลิ้มพรจิตรวิไล. เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51 ฉบับ AT89C5x/AT89Sxxxx. บริษัทอินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด. กรุงเทพฯ. 2550

# ภาคผนวก ก
# โปรแกรมควบคุมเครื่องควบคุมอุณหภูมิน้ำ

```
Onewire         bit     P2.7
flag            equ     02FH
busy            bit     flag.0
ds1820_error    bit     flag.1
champchk_1      bit     flag.2
champchk_2      bit     flag.3
dsp1            bit     p1.1
dsp2            bit     p.1.2
dsp3            bit     p1.3
set_sw          bit     p3.0
up_sw           bit     p3.1
down_sw         bit     p3.2
coolwork        bit     p2.1
hotwork         bit     p2.2
onewire_data    equ     032H
temp            equ     033H
temph           equ     034H
onewire_cut     equ     035H
dsp1_BUFFER     equ     036H
dsp2_BUFFER     equ     037H
dsp3_5or0       equ     038H
dsp1_buffer_set equ     039H
dsp2_buffer_set equ     040H
dsp3_buffer_set equ     041H
temp_set        equ     042H
temp_set_2      equ     043H
dsp2_buffer_acc equ     044H
dsp3_buffer_acc equ     045H
champ           equ     046H
champ_2         equ     047H
workdata        equ     048H
temp_work_1     equ     049H
temp_work_2     equ     050H
temp_work_ans   equ     051H
temp_work_acc   equ     052H
temp_work_relay equ     053H
champchk        equ     054H
                ORG     0000H
                mov     p0,#00000000B
                mov     dsp1_buffer_set,#2
```

```
                    mov     dsp3_buffer_set,#0
                    mov     dsp2_buffer_acc,#1
                    mov     dsp3_buffer_acc,#0
                    mov     champ_2,#0
                    clr     up_sw
                    clr     down_sw
                    clr     set_sw
                    clr     champchk_1
                    clr     champchk_2
Main:               acall   ds1820_temp_rd
                    jb      set_sw,champsw
                    acall   work
                    jb      ds1820_error,data_invalid
                    mov     a,#0aaH
                    acall   work
                    cjne    a,temp,data_valid
data_invalid:       clr     dsp1
                    mov     a,#dsp_numx
                    mov     p0,a
                    acall   delay_10ms
                    setb    dsp1
                    clr     dsp2
                    mov     a,#dsp_numx
                    mov     p0,a
                    acall   delay_10ms
                    setb    dsp2
                    clr     dsp3
                    mov     a,#dsp_numx
                    mov     p0,a
                    setb    p0.0
                    acall   delay_10ms
                    setb    dsp3
                    ajmp    main
champsw:            ajmp    champ_sw
data_valid:         mov     a,temp
                    clr     c
                    rrc     a
                    jc      five
                    mov     dsp3_5or0,#0
                    ajmp    show
```

```
                ajmp    show
show:           mov     r4,#5
                mov     b,#10
                div     ab
                mov     dsp1_buffer,a
                mov     dsp2_buffer,b
show_1:         mov     a,dsp1_buffer
                mov     dptr,#dsp_num0
                movc    a,@a+dptr
                mov     p0,a
                clr     dsp1
                acall   delay_750us
                setb    dsp1
                mov     a,dsp2_buffer
                mov     dptr,#dsp_num0
                movc    a,@a+dptr
                mov     p0,a
                setb    p0.7
                clr     dsp2
                acall   delay_750us
                setb    dsp2
                mov     a,dsp3_5or0
                mov     dptr,#dsp_num0
                movc    a,@a+dptr
                mov     p0,a
                clr     dsp3
                acall   delay_750us
                setb    dsp3
                djnz    r4,show_1
                ajmp    main
ds1820_temp_rd: clr     ds1820_error
                acall   ds1820_rst
                acall   ds1820_pres
                mov     onewire_data,#0CCH
                acall   ds1820_wr
                mov     onewire_data,#044H
                acall   ds1820_wr
                setb    onewire
                acall   delay_750us
                acall   ds1820_rst
```

```
                        mov     onewire_data,#0CCH
                        acall   ds1820_wr
                        mov     onewire_data,#0BEH
                        acall   ds1820_wr
                        acall   ds1820_rd
                        mov     temp,onewire_data
                        acall   ds1820_rd
                        mov     temph,onewire_data
ds1820_temp_clr:        acall   ds1820_rst
                        acall   ds1820_pres
ds1820_temp_exit:       ret
ds1820_rd:              mov     onewire_cut,#8
                        clr     a
ds1820_rd_loop:         clr     onewire
                        nop
                        setb    onewire
                        nop
                        nop
                        mov     c,onewire
                        acall   onewire_delay
                        rrc     a
                        djnz    onewire_cut,ds1820_rd_loop
                        mov     onewire_data,a
                        ret
ds1820_wr:              mov     onewire_cut,#8
                        mov     a,onewire_data
ds1820_wr_loop:         rrc     a
                        jnc     ds1820_wr_l
                        clr     onewire
                        nop
                        nop
                        setb    onewire
                        acall   onewire_delay
                        ajmp    ds1820_wr_nx
ds1820_wr_l:            clr     onewire
                        acall   onewire_delay
                        setb    onewire
                        nop
                        nop
ds1820_wr_nx:           djnz    onewire_cut,ds1820_wr_loop
```

```
ds1820_rst:     clr     onewire
                acall   delay_750us
                setb    onewire
                mov     onewire_cut,#11
                djnz    onewire_cut,$
                ret
ds1820_pres:    mov     onewire_cut,#55
ds1820_pres_1:  jnb     onewire,ds1820_pres_2
                djnz    onewire_cut,ds1820_pres_1
                setb    ds1820_error
                ret
ds1820_pres_2:  mov     onewire_cut,#55
ds1820_pres_3:  jb      onewire,ds1820_pres_4
                djnz    onewire_cut,ds1820_pres_3
                setb    ds1820_error
ds1820_pres_4:  setb    onewire
                acall   onewire_delay
                clr     busy
                ret
loop:           mov     champ,#0
                ajmp    main
champ_sw:       acall   delay_10ms
                jnb     set_sw,sw_mode
                ajmp    champ_sw
sw_mode:        xch     a,champ
                cjne    a,#0,sw_mode_1
                mov     a,#1
                xch     a,champ
                clr     hotwork
                clr     coolwork
                clr     champchk_2
                clr     champchk_1
                ajmp    setmode
sw_mode_1:      cjne    a,#1,loop
                mov     a,#2
                xch     a,champ
                ajmp    setmode2
keypressed_2:   jnb     up_sw,upmode
                ajmp    keypressed_2
keypressed_3:   jnb     down_sw,downmode
```

```
setmode:    mov     r4,#06DH
setmode_1:  mov     a,dsp1_buffer_set
            mov     dptr,#dsp_num0
            movc    a,@a+dptr
            mov     p0,a
            clr     dsp1
            acall   delay_750us
            setb    dsp1
            mov     a,dsp2_buffer_set
            mov     dptr,#dsp_num0
            movc    a,@a+dptr
            mov     p0,a
            setb    p0.7
            clr     dsp2
            acall   delay_750us
            setb    dsp2
            mov     a,dsp3_buffer_set
            mov     dptr,#dsp_num0
            movc    a,@a+dptr
            mov     p0,a
            clr     dsp3
            acall   delay_750us
            setb    dsp3
            jb      up_sw,keypressed_2
            jb      down_sw,keypressed_3
            jb      set_sw,champ_sw
            djnz    r4,setmode_1
            acall   delay_100ms
            ajmp    setmode_1
upmode:     mov     a,dsp3_buffer_set
            cjne    a,#0,upmode_1
            mov     dsp3_buffer_set,#5
            ajmp    upmode_2
upmode_1:   mov     dsp3_buffer_set,#0
            mov     a,dsp1_buffer_set
            mov     b,#10
            mul     ab
            add     a,dsp2_buffer_set
            add     a,#1
            mov     b,#10
```

```
                    mov     dsp1_buffer_set,a
                    mov     dsp2_buffer_set,b
upmode_2:           acall   delay_10ms
                    jnb     up_sw,setmode
                    ajmp    upmode_2
downmode:           mov     a,dsp3_buffer_set
                    cjne    a,#5,downmode_1
                    mov     dsp3_buffer_set,#0
                    ajmp    downmode_2
downmode_1:         mov     dsp3_buffer_set,#5
                    mov     a,dsp1_buffer_set
                    mov     b,#10
                    mul     ab
                    add     a,dsp2_buffer_set
                    dec     a
                    mov     b,#10
                    div     ab
                    mov     dsp1_buffer_set,a
                    mov     dsp2_buffer_set,b
downmode_2:         acall   delay_10ms
                    jnb     down_sw,champd_j
                    ajmp    downmode_2
champd_j:           ajmp    setmode
setmode2:           mov     r4,#06DH
setmode2_1:         mov     a,dsp2_buffer_acc
                    mov     dptr,#dsp_num0
                    movc    a,@a+dptr
                    mov     p0,a
                    setb    p0.7
                    clr     dsp2
                    acall   delay_750us
                    setb    dsp2
                    mov     a,dsp3_buffer_acc
                    mov     dptr,#dsp_num0
                    movc    a,@a+dptr
                    mov     p0,a
                    clr     dsp3
                    acall   delay_750us
                    setb    dsp3
                    jb      up_sw,upmode2
```

```
                djnz    r4,setmode2_1
                acall   delay_100ms
                ajmp    setmode2_1
champ_j:        ajmp    champ_sw
upmode2:        acall   delay_10ms
                jnb     up_sw,upmode2_1
                ajmp    upmode2
upmode2_1:      xch     a,champ_2
                cjne    a,#0,upmode2_2
                mov     a,#1
                mov     dsp2_buffer_acc,#1
                mov     dsp3_buffer_acc,#0
                xch     a,champ_2
                ajmp    setmode2
upmode2_2:      cjne    a,#1,upmode2_3
                mov     a,#2
                mov     dsp2_buffer_acc,#1
                mov     dsp3_buffer_acc,#5
                xch     a,champ_2
                ajmp    setmode2
upmode2_3:      mov     a,#0
                mov     dsp2_buffer_acc,#2
                mov     dsp3_buffer_acc,#0
                xch     a,champ_2
                ajmp    setmode2
onewire_delay:  mov     r6,#012H
onewire_delay_1: nop
                nop
                djnz    r6,onewire_delay_1
                ret
delay_1s:       mov     r5,#100
delay_1s_1:     acall   delay_10ms
                djnz    r5,delay_1s_1
                ret
delay_100ms:    mov     r5,#50
delay_100ms_1:  acall   delay_10ms
                djnz    r5,delay_100ms_1
                ret
delay_10ms:     mov     r7,#010
delay_10ms_1:   mov     r6,#0E6H
```

```
                    nop
                    djnz    r6,delay_10ms_2
                    djnz    r7,delay_10ms_1
                    ret
delay_750us:        mov     r6,#229
delay_750us_2:      nop
                    djnz    r6,delay_750us_2
                    ret
work:               jb      champchk_1,gotowork_5
                    jb      champchk_2,gotowork_5
                    ajmp    gotowork
gotochamp_j3:       ajmp    champ_j3
gotowork_5:         ajmp    work_5
gotowork:           mov     a,dsp2_buffer_acc
                    mov     b,#10
                    mul     ab
                    add     a,dsp3_buffer_acc
                    mov     temp_work_acc,a
work_1:             mov     a,dsp1_buffer_set
                    mov     b,#10
                    mul     ab
                    add     a,dsp2_buffer_set
                    mov     temp_work_1,a
                    mov     a,dsp1_buffer
                    mov     b,#10
                    mul     ab
                    add     a,dsp2_buffer
                    mov     temp_work_2,a
                    mov     a,temp_work_1
                    clr     c
                    cjne    a,temp_work_2,more
                    clr     coolwork
                    clr     hotwork
                    ajmp    champ_j3
more:               jc      less
                    mov     temp_work_relay,#1
                    mov     a,temp_work_1
                    subb    a,temp_work_2
                    mov     b,#10
                    mul     ab
```

```
                mov     a,dsp3_buffer_set
                cjne    a,dsp3_5or0,more_less
                ajmp    work_2
more_more:      jc      more_less
                mov     a,temp_work_ans
                add     a,#5
                mov     temp_work_ans,a
                ajmp    work_2
more_less:      clr     c
                mov     a,temp_work_ans
                subb    a,#5
                mov     temp_work_ans,a
                ajmp    work_2
less:           clr     c
                mov     temp_work_relay,#2
                mov     a,temp_work_2
                subb    a,temp_work_1
                mov     b,#10
                mul     ab
                mov     temp_work_ans,a
                mov     a,dsp3_5or0
                cjne    a,dsp3_buffer_set,less_more
                ajmp    work_2
less_more:      jc      less_less
                mov     a,temp_work_ans
                add     a,#5
                mov     temp_work_ans,a
                ajmp    work_2
less_less:      clr     c
                mov     a,temp_work_ans
                subb    a,#5
                mov     temp_work_ans,a
                ajmp    work_2
work_2:         mov     a,temp_work_ans
                cjne    a,temp_work_acc,work_3
                ajmp    work_relay
work_3:         jc      work_4
                ajmp    work_relay
work_4:         ajmp    champ_j3
work_relay:     mov     a,temp_work_relay
```

```
                    clr     hotwork
                    clr     champchk_1
                    setb    coolwork
                    setb    champchk_2
                    ajmp    champ_j3
work_hotwork:       clr     coolwork
                    clr     champchk_2
                    setb    hotwork
                    setb    champchk_1
                    ajmp    champ_j3
work_5:             mov     a,dsp1_buffer_set
                    cjne    a,dsp1_BUFFER,champ_j3
                    mov     a,dsp2_buffer_set
                    cjne    a,dsp2_BUFFER,champ_j3
                    mov     a,dsp3_buffer_set
                    cjne    a,dsp3_5or0,champ_j3
                    clr     hotwork
                    clr     coolwork
                    clr     champchk_2
                    clr     champchk_1
champ_j3:           ret
;segment
dsp_num0:           db      00111111B
dsp_num1:           db      00000110B
dsp_num2:           db      01011011B
dsp_num3:           db      01001111B
dsp_num4:           db      01100110B
dsp_num5:           db      01101101B
dsp_num6:           db      01111101B
dsp_num7:           db      00000111B
dsp_num8:           db      01111111B
dsp_num9:           db      01101111B
dsp_numx:           db      01000000B
end
```

# ภาคผนวก ข

# รายละเอียดของไอซีหมายเลข DS18S20

19-5474; Rev 8/10

DALLAS SEMICONDUCTOR MAXIM

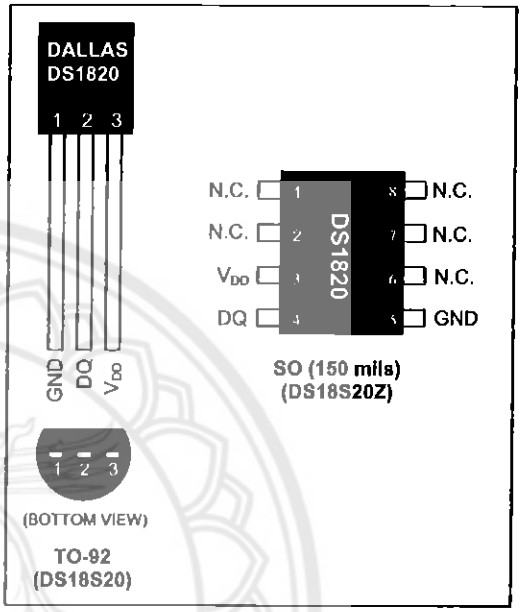
# DS18S20
# High-Precision 1-Wire Digital Thermometer

www.maxim-ic.com

## 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
- Multidrop 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°C (-67°F to +257°F)
- ±0.5°C Accuracy from -10°C to +85°C
- 9-Bit Thermometer Resolution
- Converts Temperature in 750ms (max)
- User-Definable Nonvolatile (NV) Alarm Settings
- Alarm Search Command Identifies and Addresses Devices Whose Temperature is Outside Programmed Limits (Temperature Alarm Condition)
- Applications Include Thermostatic Controls, Industrial Systems, Consumer Products, Thermometers, or Any Thermally Sensitive System

## PIN CONFIGURATIONS




## DESCRIPTION

The DS18S20 digital thermometer provides 9-bit Celsius 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°C and is accurate to ±0.5°C over the range of –10°C to +85°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.

*1-Wire is a registered trademark of Maxim Integrated Products, Inc.*

## ORDERING INFORMATION

| PART | TEMP RANGE | PIN-PACKAGE |
|---|---|---|
| DS18S20 | –55°C to +125°C | 3 TO-92 |
| DS18S20+ | –55°C to +125°C | 3 TO-92 |
| DS18S20/T&R | –55°C to +125°C | 3 TO-92 (2000 Piece) |
| DS18S20+T&R | –55°C to +125°C | 3 TO-92 (2000 Piece) |
| DS18S20-SL/T&R | –55°C to +125°C | 3 TO-92 (2000 Piece)* |
| DS18S20-SL+T&R | –55°C to +125°C | 3 TO-92 (2000 Piece)* |
| DS18S20Z | –55°C to +125°C | 8 SO |
| DS18S20Z+ | –55°C to +125°C | 8 SO |
| DS18S20Z/T&R | –55°C to +125°C | 8 SO (2500 Piece) |
| DS18S20Z+T&R | –55°C to +125°C | 8 SO (2500 Piece) |

*+Denotes a lead(Pb)-free/RoHS-compliant package. A "+" appears on the top mark of lead(Pb)-free packages.*
*T&R = Tape and reel.*
**TO-92 packages in tape and reel can be ordered with straight or formed leads. Choose "SL" for straight leads. Bulk TO-92 orders are straight leads only.*

## PIN DESCRIPTION

| PIN | | NAME | FUNCTION |
|---|---|---|---|
| TO-92 | SO | | |
| 1 | 5 | GND | Ground |
| 2 | 4 | DQ | Data Input/Output. Open-drain 1-Wire interface pin. Also provides power to the device when used in parasite power mode (see the *Powering the DS18S20* section.) |
| 3 | 3 | $V_{DD}$ | Optional $V_{DD}$. $V_{DD}$ must be grounded for operation in parasite power mode. |
| — | 1, 2, 6, 7, 8 | N.C. | No Connection |

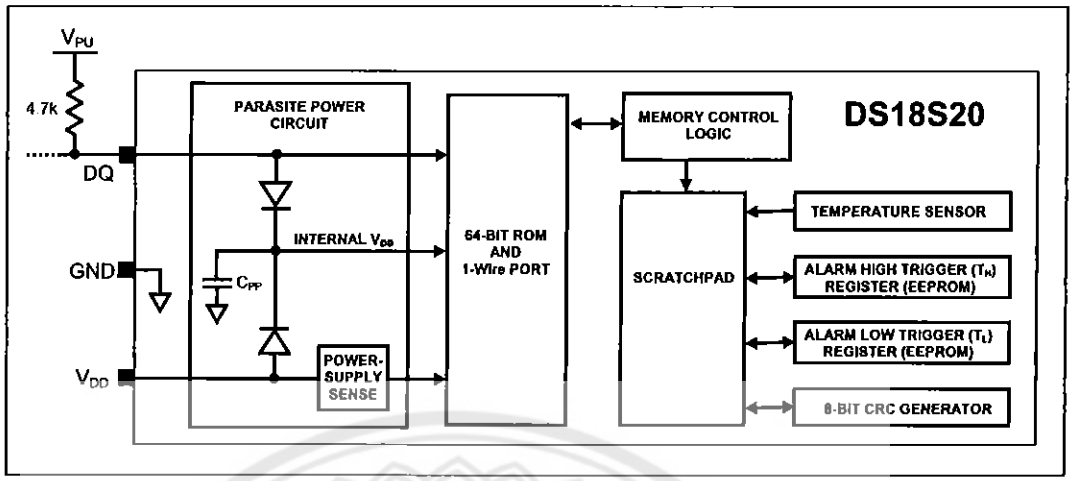
## OVERVIEW

Figure 1 shows a block diagram of the DS18S20, and pin descriptions are given in the *Pin Description* table. 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 Maxim's 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.

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 $V_{DD}$.

**Figure 1. DS18S20 Block Diagram**




## 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.

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 1 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:

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

**Figure 2. Temperature Register Format**

| | 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 |

S = SIGN

**Table 1. Temperature/Data Relationship**

| TEMPERATURE (°C) | DIGITAL OUTPUT (BINARY) | DIGITAL OUTPUT (HEX) |
|---|---|---|
| +85.0* | 0000 0000 1010 1010 | 00AAh |
| +25.0 | 0000 0000 0011 0010 | 0032h |
| +0.5 | 0000 0000 0000 0001 | 0001h |
| 0 | 0000 0000 0000 0000 | 0000h |
| -0.5 | 1111 1111 1111 1111 | FFFFh |
| -25.0 | 1111 1111 1100 1110 | FFCEh |
| -55.0 | 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.

**Figure 3. $T_H$ and $T_L$ Register Format**

| 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 measured temperature is lower than or equal to $T_L$ or higher than $T_H$, 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.

# ภาคผนวก ค

# รายละเอียดของทรานซิสเตอร์หมายเลข BC327

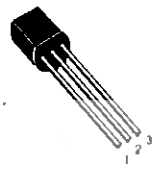
# BC327-25
# BC327-40

## SMALL SIGNAL PNP TRANSISTORS

**PRELIMINARY DATA**

| Type | Marking |
|---|---|
| BC327-25 | BC327-25 |
| BC327-40 | BC327-40 |

- SILICON EPITAXIAL PLANAR PNP TRANSISTORS
- TO-92 PACKAGE SUITABLE FOR THROUGH-HOLE PCB ASSEMBLY
- THE NPN COMPLEMENTARY TYPES ARE BC337-25 AND BC337-40 RESPECTIVELY

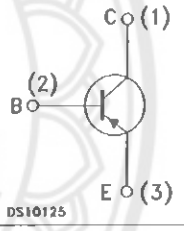



TO-92

### APPLICATIONS

- WELL SUITABLE FOR TV AND HOME APPLIANCE EQUIPMENT
- SMALL LOAD SWITCH TRANSISTORS WITH HIGH GAIN AND LOW SATURATION VOLTAGE

**INTERNAL SCHEMATIC DIAGRAM**




### ABSOLUTE MAXIMUM RATINGS

| Symbol | Parameter | Value | Unit |
|---|---|---|---|
| $V_{CBO}$ | Collector-Base Voltage ($I_E$ = 0) | -50 | V |
| $V_{CEO}$ | Collector-Emitter Voltage ($I_B$ = 0) | -45 | V |
| $V_{EBO}$ | Emitter-Base Voltage ($I_C$ = 0) | -5 | V |
| $I_C$ | Collector Current | -0.5 | A |
| $I_{CM}$ | Collector Peak Current | -1 | A |
| $P_{tot}$ | Total Dissipation at $T_C$ = 25 °C | 625 | mW |
| $T_{stg}$ | Storage Temperature | -65 to 150 | °C |
| $T_j$ | Max. Operating Junction Temperature | 150 | °C |

## THERMAL DATA

| | | | | |
|---|---|---|---|---|
| $R_{thj\text{-}amb}$ • | Thermal Resistance Junction-Ambient | Max | 200 | °C/W |
| $R_{thj\text{-}case}$ • | Thermal Resistance Junction-Case | Max | 83.3 | °C/W |

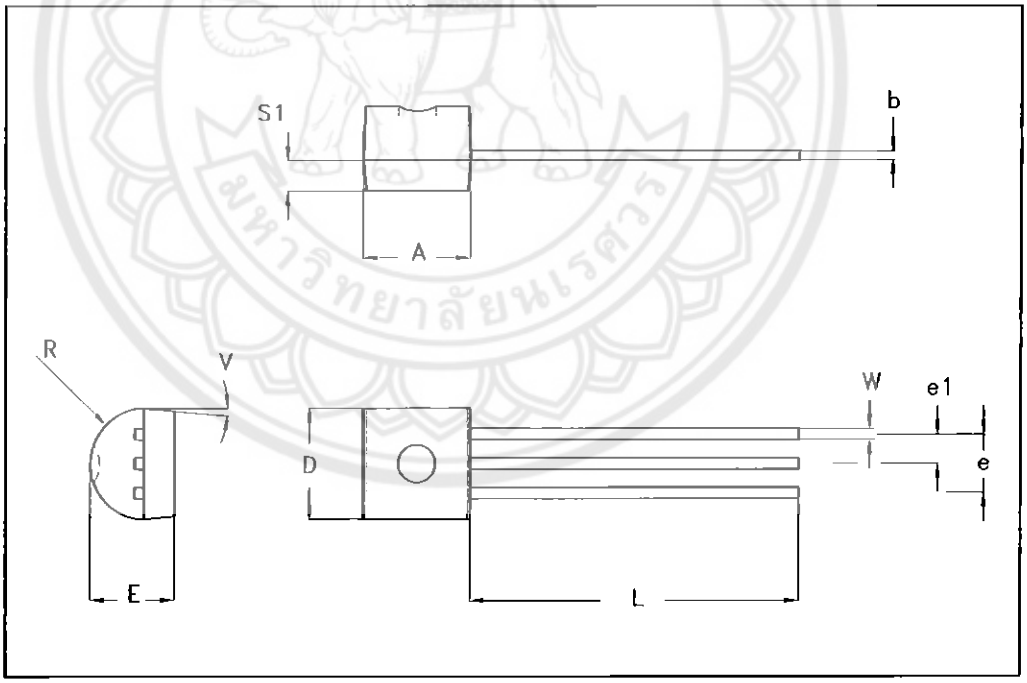
## ELECTRICAL CHARACTERISTICS ($T_{case}$ = 25 °C unless otherwise specified)

| Symbol | Parameter | Test Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $I_{CBO}$ | Collector Cut-off Current ($I_E$ = 0) | $V_{CB}$ = -20 V<br>$V_{CB}$ = -20 V $T_C$ = 150 °C | | | -100<br>-5 | nA<br>µA |
| $I_{EBO}$ | Emitter Cut-off Current ($I_C$ = 0) | $V_{EB}$ = -5 V | | | -100 | nA |
| $V_{(BR)CBO}$ | Collector-Base Breakdown Voltage ($I_E$ = 0) | $I_C$ = -10 µA | -50 | | | V |
| $V_{(BR)CEO}$* | Collector-Emitter Breakdown Voltage ($I_B$ = 0) | $I_C$ = -10 mA | -45 | | | V |
| $V_{(BR)EBO}$ | Emitter-Base Breakdown Voltage ($I_C$ = 0) | $I_E$ = -10 µA | -5 | | | V |
| $V_{CE(sat)}$* | Collector-Emitter Saturation Voltage | $I_C$ = -500 mA $I_B$ = -50 mA | | | -0.7 | V |
| $V_{BE(on)}$* | Base-Emitter On Voltage | $I_C$ = -500 mA $V_{CE}$ = -1 V | | | -1.2 | V |
| $h_{FE}$* | DC Current Gain | $I_C$ = -100 mA $V_{CE}$ = -1 V<br>for **BC327-25**<br>for **BC327-40** | <br>160<br>250 | | <br>400<br>600 | |
| $f_T$ | Transition Frequency | $I_C$ = -10mA $V_{CE}$ = -5 V f = 100MHz | 80 | | | MHz |
| $C_{CBO}$ | Collector-Base Capacitance | $I_E$ = 0 $V_{CB}$ = -10 V f = 1 MHz | | 10 | | pF |

* Pulsed: Pulse duration = 300 µs, duty cycle ≤ 2 %

## TO-92 MECHANICAL DATA

| DIM. | mm | | | inch | | |
|---|---|---|---|---|---|---|
| | MIN. | TYP. | MAX. | MIN. | TYP. | MAX. |
| A | 4.32 | | 4.95 | 0.170 | | 0.195 |
| b | 0.36 | | 0.51 | 0.014 | | 0.020 |
| D | 4.45 | | 4.95 | 0.175 | | 0.194 |
| E | 3.30 | | 3.94 | 0.130 | | 0.155 |
| e | 2.41 | | 2.67 | 0.095 | | 0.105 |
| e1 | 1.14 | | 1.40 | 0.045 | | 0.055 |
| L | 12.70 | | 15.49 | 0.500 | | 0.609 |
| R | 2.16 | | 2.41 | 0.085 | | 0.094 |
| S1 | 1.14 | | 1.52 | 0.045 | | 0.059 |
| W | 0.41 | | 0.56 | 0.016 | | 0.022 |
| V | 4 degree | | 6 degree | 4 degree | | 6 degree |



# ภาคผนวก ง

# รายละเอียดของทรานซิสเตอร์หมายเลข BC337

FAIRCHILD
SEMICONDUCTOR

# BC337/338

## Switching and Amplifier Applications

- Suitable for AF-Driver stages and low power output stages
- Complement to BC327/BC328

TO-92

1. Collector 2. Base 3. Emitter

## NPN Epitaxial Silicon Transistor

### Absolute Maximum Ratings $T_a$=25°C unless otherwise noted

| Symbol | Parameter | Value | Units |
|---|---|---|---|
| $V_{CES}$ | Collector-Emitter Voltage<br>: BC337<br>: BC338 | <br>50<br>30 | <br>V<br>V |
| $V_{CEO}$ | Collector-Emitter Voltage<br>: BC337<br>: BC338 | <br>45<br>25 | <br>V<br>V |
| $V_{EBO}$ | Emitter-Base Voltage | 5 | V |
| $I_C$ | Collector Current (DC) | 800 | mA |
| $P_C$ | Collector Power Dissipation | 625 | mW |
| $T_J$ | Junction Temperature | 150 | °C |
| $T_{STG}$ | Storage Temperature | -55 ~ 150 | °C |

### Electrical Characteristics $T_a$=25°C unless otherwise noted

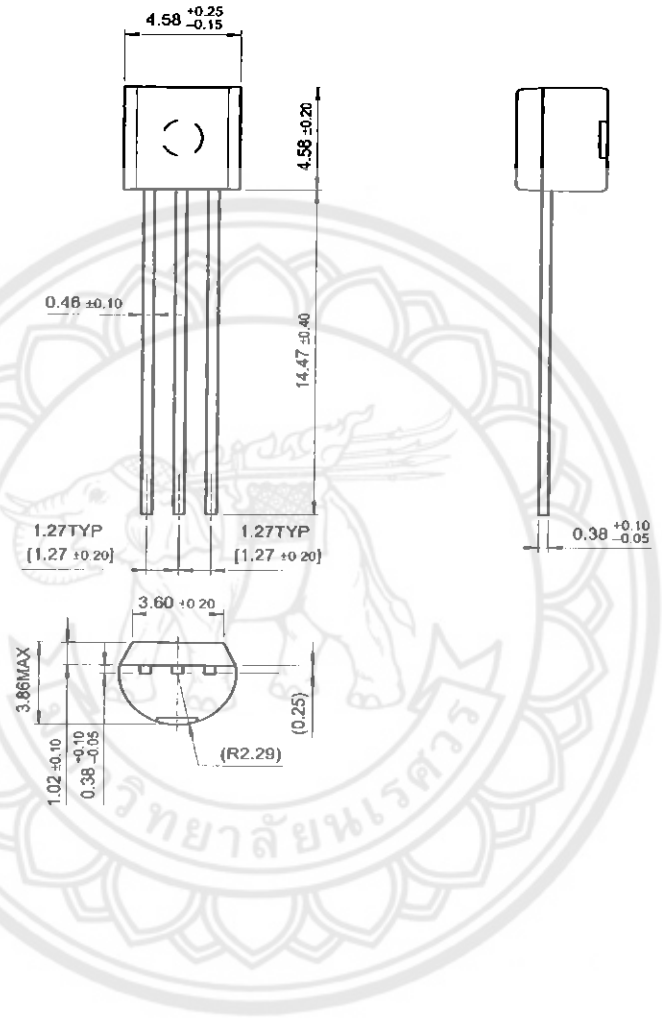
| Symbol | Parameter | Test Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $BV_{CEO}$ | Collector-Emitter Breakdown Voltage<br>: BC337<br>: BC338 | $I_C$=10mA, $I_B$=0 | <br>45<br>25 | | | <br>V<br>V |
| $BV_{CES}$ | Collector-Emitter Breakdown Voltage<br>: BC337<br>: BC338 | $I_C$=0.1mA, $V_{BE}$=0 | <br>50<br>30 | | | <br>V<br>V |
| $BV_{EBO}$ | Emitter-Base Breakdown Voltage | $I_E$=0.1mA, $I_C$=0 | 5 | | | V |
| $I_{CES}$ | Collector Cut-off Current<br>: BC337<br>: BC338 | <br>$V_{CE}$=45V, $I_B$=0<br>$V_{CE}$=25V, $I_B$=0 | <br> | <br>2<br>2 | <br>100<br>100 | <br>nA<br>nA |
| $h_{FE1}$<br>$h_{FE2}$ | DC Current Gain | $V_{CE}$=1V, $I_C$=100mA<br>$V_{CE}$=1V, $I_C$=300mA | 100<br>60 | | 630 | |
| $V_{CE}$ (sat) | Collector-Emitter Saturation Voltage | $I_C$=500mA, $I_B$=50mA | | | 0.7 | V |
| $V_{BE}$ (on) | Base Emitter On Voltage | $V_{CE}$=1V, $I_C$=300mA | | | 1.2 | V |
| $f_T$ | Current Gain Bandwidth Product | $V_{CE}$=5V, $I_C$=10mA, f=50MHz | | 100 | | MHz |
| $C_{ob}$ | Output Capacitance | $V_{CB}$=10V, $I_E$=0, f=1MHz | | 12 | | pF |

### $h_{FE}$ Classification

| Classification | 16 | 25 | 40 |
|---|---|---|---|
| $h_{FE1}$ | 100 ~ 250 | 160 ~ 400 | 250 ~ 630 |
| $h_{FE2}$ | 60- | 100- | 170- |

## Package Dimensions

TO-92




Dimensions in Millimeters

## TRADEMARKS

The following are registered and unregistered trademarks Fairchild Semiconductor owns or is authorized to use and is not intended to be an exhaustive list of all such trademarks.

| | | | | |
|---|---|---|---|---|
| ACEx™ | FACT™ | ImpliedDisconnect™ | PACMAN™ | SPM™ |
| ActiveArray™ | FACT Quiet series™ | ISOPLANAR™ | POP™ | Stealth™ |
| Bottomless™ | FAST® | LittleFET™ | Power247™ | SuperSOT™-3 |
| CoolFET™ | FASTr™ | MicroFET™ | PowerTrench® | SuperSOT™-6 |
| *CROSSVOLT*™ | FRFET™ | MicroPak™ | QFET™ | SuperSOT™-8 |
| DOME™ | GlobalOptoisolator™ | MICROWIRE™ | QS™ | SyncFET™ |
| EcoSPARK™ | GTO™ | MSX™ | QT Optoelectronics™ | TinyLogic™ |
| $E^2CMOS$™ | HiSeC™ | MSXPro™ | Quiet Series™ | TruTranslation™ |
| EnSigna™ | $I^2C$™ | OCX™ | RapidConfigure™ | UHC™ |
| Across the board. Around the world.™ | | OCXPro™ | RapidConnect™ | UltraFET® |
| The Power Franchise™ | | OPTOLOGIC® | SILENT SWITCHER® | VCX™ |
| Programmable Active Droop™ | | OPTOPLANAR™ | SMART START™ | |

## DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

## LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF FAIRCHILD SEMICONDUCTOR CORPORATION.

As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, or (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in significant injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

## PRODUCT STATUS DEFINITIONS

### Definition of Terms

| Datasheet Identification | Product Status | Definition |
|---|---|---|
| Advance Information | Formative or In Design | This datasheet contains the design specifications for product development. Specifications may change in any manner without notice. |
| Preliminary | First Production | This datasheet contains preliminary data, and supplementary data will be published at a later date. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design. |
| No Identification Needed | Full Production | This datasheet contains final specifications. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design. |
| Obsolete | Not In Production | This datasheet contains specifications on a product that has been discontinued by Fairchild semiconductor. The datasheet is printed for reference information only. |

# ประวัติผู้ดำเนินโครงงาน

ชื่อ นายณัฐพล ชอบจิตต์

ภูมิลำเนา 59/57 หมู่บ้านถาวรกาญจน์ หมู่ 3 ต. ท่าทอง อ. เมือง จ. พิษณุโลก

ประวัติการศึกษา

– จบระดับมัธยมศึกษาจากโรงเรียนพิษณุโลกพิทยาคม จ.พิษณุโลก

– ปัจจุบันกำลังศึกษาระดับปริญญาตรี ชั้นปีที่ 4 สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

Email: sanimclub@hotmail.com

ชื่อ นายธีรพงศ์ มีแม่นวิทย์

ภูมิลำเนา 696/2 ถ.บุษบา ต.ในเมือง อ.เมือง จ.พิจิตร

ประวัติการศึกษา

– จบระดับมัธยมศึกษาจากโรงเรียนพิจิตรพิทยาคม จ.พิจิตรพิทยาคม

– ปัจจุบันกำลังศึกษาระดับปริญญาตรี ชั้นปีที่ 4 สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

Email: xmagehero@hotmail.com